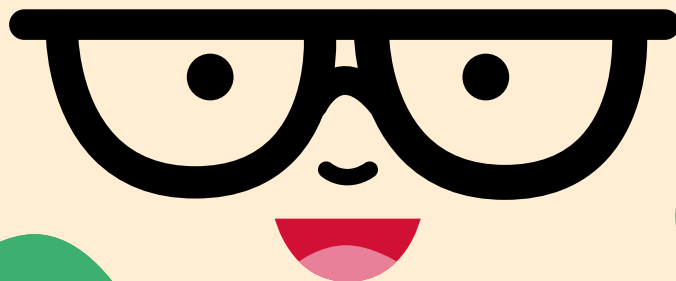


kluraklubb

Matematik och programmering

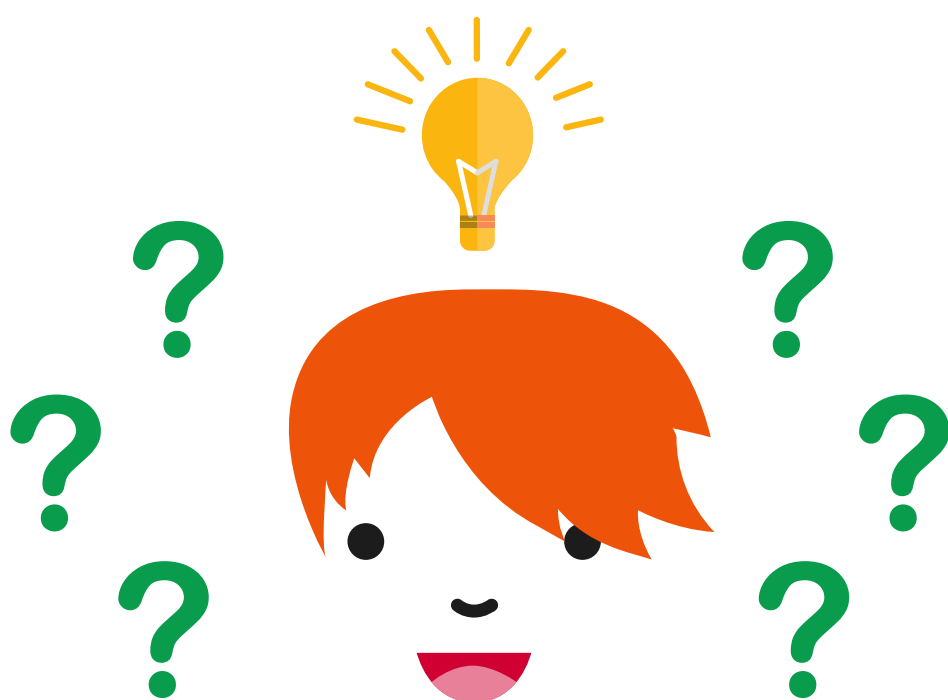


Handledarguide



kluraklubb

Matematik och programmering



Handledarguide



Första upplagan (2016) sammanställd av

Helsingfors universitets LUMA-center

Sumu Virpi, koordinator, centret Linkki

Tuominen Jenna, koordinator, centret Linkki

Räsänen Jenni, koordinator, centret Summamutikka

Helmet-biblioteken

Liljelund Lilja, Esbo stadsbibliotek

Gros Mikael, Esbo stadsbibliotek

Kaukoniemi Henna, Vanda stadsbibliotek

Terrihauta Pia, Helsingfors stadsbibliotek

Producent projektet Kluraklubb

Förläggare Helsingfors stadsbibliotek

Undervisnings- och kulturministeriet

Layout Hautala Marja, Muuks Creative

Illustrering Helsingfors universitets LUMA-center och
Hautala Marja

Översättning Nordström Ann

Den andra uppdaterade upplagan (2019) sammanställd av

Sumu Virpi, koordinator, centret Linkki

Tuominen Jenna, koordinator, centret Linkki

Förläggare Helsingfors universitets vetenskapsutbildningscenter

Helsingfors 2019

ISBN 978-951-51-4986-2 (pdf)

DOI <https://doi.org/10.31885/9789515149862>

HELMET



Undervisnings- och
kulturministeriet



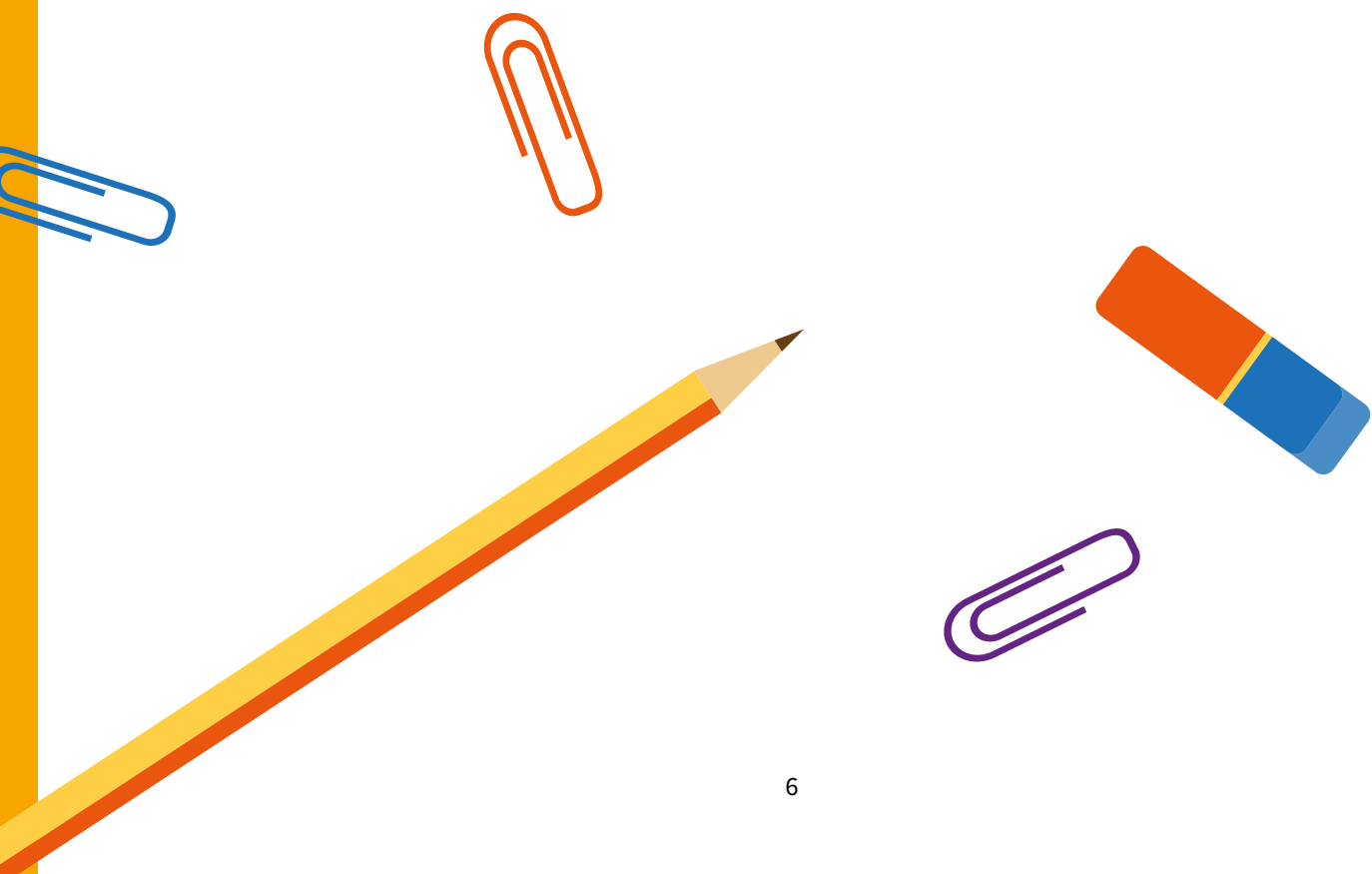


Innehållsförteckning

Inledning	6
Praktiskt genomförande av verkstäderna.....	8
Inledning till matematik-uppgifterna	12
Inledning till programmerings-uppgifterna.....	14
Programmeringsdelens framskridande.....	15
Scratch-programmeringsmiljön.....	16
Handledning av programmeringen	21
Anvisningar för verkstäderna	26
Första verkstaden: Juveltjuven	29
Andra verkstaden: Havsäventyret.....	37
Tredje verkstaden: Astronomerna	47
Fjärde verkstaden: Detektiverna	57
Femte verkstaden: Jordenruntresan	69
Fler problem	78
Bilagor	80



Inledning



Du håller i handledarguiden för Kluraklubb-verkstäderna. Kluraklubb är en funktionell verkstadshelhet om matematik och programmering för 9–13-åringar. Guidens uppgifter är ett enkelt och intressant sätt att närma sig matematik och programmering och det krävs ingen tidigare erfarenhet för att kunna handleda dem.

I guiden finns anvisningar för fem verkstäder. En verkstads tänkta längd är två timmar varav den ena delen används för de matematiska problemen och den andra för att programmera det egna spelet. Du kan också dela in verkstadsuppgifterna på ett sätt som passar dig och gruppen, till exempel en timmes verkstäder, varvid verkstadens tema varannan vecka är matematik och varannan programmering.

Kluraklubb startade som ett gemensamt projekt mellan Helmet-biblioteken, dvs. stadsbiblioteken i Esbo, Helsingfors, Grankulla och Vanda, samt Helsingfors universitets LUMA-center som finansierades av undervisnings- och kulturministeriet. Från Helsingfors universitets LUMA-center deltog i projektet inlärningscentret för matematik Summamutikka och resurscentret för datateknik Linkki. Bibliotek från alla Helmet-kommuner deltog.

Vi önskar er glada stunder med matematik och programmering!

Helsingfors, december 2016

Arbetsgruppen för projektet Kluraklubb

Andra uppdaterade upplagan

Uppdateringen av Scratch-programmeringsmiljön innebar att Kluraklubbs handledarguide behövde uppdateras. Denna handledarguide motsvarar Scratch version 3.0. Det är möjligt att använda den nya Scratch också med bl.a. surfplattor, men vi tycker det fortfarande är lättare att använda Scratch på en dator där man bl.a. har en större skärm och en mus. Därför tar vi inte hänsyn till pekskärmar i guiden. Om man använder den äldre Scratch, rekommenderar vi den tidigare versionen av guiden.

Vi önskar er givande programmeringsstunder med den nya, uppdaterade guiden!

Helsingfors, december 2019

Virpi Sumu och Jenna Tuominen

Praktiskt genomförande av verkstäderna

FÖRE VERKSTADEN

Det är bra att börja **marknadsföra** det praktiska genomförandet av Kluraklubb-verkstäderna i god tid. Utöver bibliotekens egna kanaler har man inom Kluraklubb-projektet konstaterat att det är bra praxis att skicka marknadsföringsmeddelanden via skolor. Då får även föräldrarna veta om verkstäderna.

Det lönar sig att använda **förhandsanmälning** för verkstäderna så att det är möjligt att reservera tillräckligt med material och utrustningar. Detta binder också i viss mån deltagarna, även om alla som anmält sig inte alltid kommer.

Det rekommenderade antalet deltagare i verkstäderna är **högst 20 deltagare** så att arbetet i verkstaden är meningsfullt och handledarnas tid räcker till för alla frågor. Det rekommenderas att **minst två biblioteksanställda som är förtrogna med materialen** är handledare. Arbetsfördelningen kan till exempel vara sådan att den ena handledaren ansvarar för matematikuppgifterna och den andra för programmeringsdelen. Även till exempel frivilliga unga kan vara hjälphandledare.

Kluraklubb-verkstäderna kan hållas i en öppen eller sluten lokal. Alla deltagare måste emellertid se bilden som handledaren projicerar från sin dator väl så utrymmet ska möbleras enligt detta.

UNDER VERKSTADEN

I början av verkstäderna, före det egentliga programmet, är det bra att använda en liten stund till att bekanta sig med de andra i gruppen, eftersom deltagarna kan vara fullständigt främmande för varandra. Till exempel på Mannerheims Barnskyddsförbunds webbplats finns det mycket bra lektipps (på finska) http://www.mll.fi/vanhempainnetti/lasten_leikit/.


Verkstädernas program består av en matematik- och programmeringsdel. Man kan välja i vilken ordning man behandlar ämnena, det spelar alltså ingen roll om man inleder verkstaden med matematik eller programmering. Säkerställ emellertid att datorerna bara används under programmeringsdelen så att de inte stör det övriga arbetet.

Det lönar sig att börja göra uppgifterna genom att läsa högt **berättelsen som leder till temat**, den finns i guiden i början av beskrivningen av varje verkstad. Lämplig litteratur att tipsa om för deltagarna i verkstaden har även sammanställts i samband med varje beskrivning.

Utöver denna guide finns det problem och tilläggsmaterial för verkstäderna på webbplatsen <https://pulmaario.luma.fi/>. Där finns bland annat uppgifter för verkstäderna, instruktionsvideor, modellspel för programmeringsavsnittet för varje verkstad, modeller som kan skrivas ut för verkstadsuppgifterna i matematik samt introduktioner till uppgifterna som förklarar de matematiska principerna och är mer omfattande än anvisningarna i denna guide.

Material och verktyg

En verkstadshelhet har planerats så att det är lätt att genomföra verkstäderna med lite material. För att genomföra de fem verkstadshelheterna i denna guide behövs följande material:

- 
- 1 påse (500 g) torkade örter
 - 1 000 st. cocktailstickor
 - 1 rulle genomskinlig tejp
 - 1 rulle målar-tejp (bredd ca 1–2 cm)
 - 2 st. genomskinliga engångsmuggar
 - några djupa papperstallrikar eller andra kärl
 - papplåda eller annat stort kärl
 - provpåsklämmor
 - kartong
 - färgat utskriftspapper
 - pappersark A4
 - sax
 - lim
 - permanent tuschpenna
 - pennor
 - stor spelbricka (t.ex. bok eller kramdjur)
 - (passare)
 - uppgiftsmallar som kopieras till deltagarna
(modeller som bilagor till guiden)

Utöver materialet behövs för programmeringen **datorer** och **möss**, **videokanon** eller **stor skärm** samt **internetförbindelse**. Det borde finnas minst en dator och mus per par, det vill säga antalet deltagare delat med två. Till exempel för en verkstad med 20 deltagare behövs alltså 10 datorer och möss.

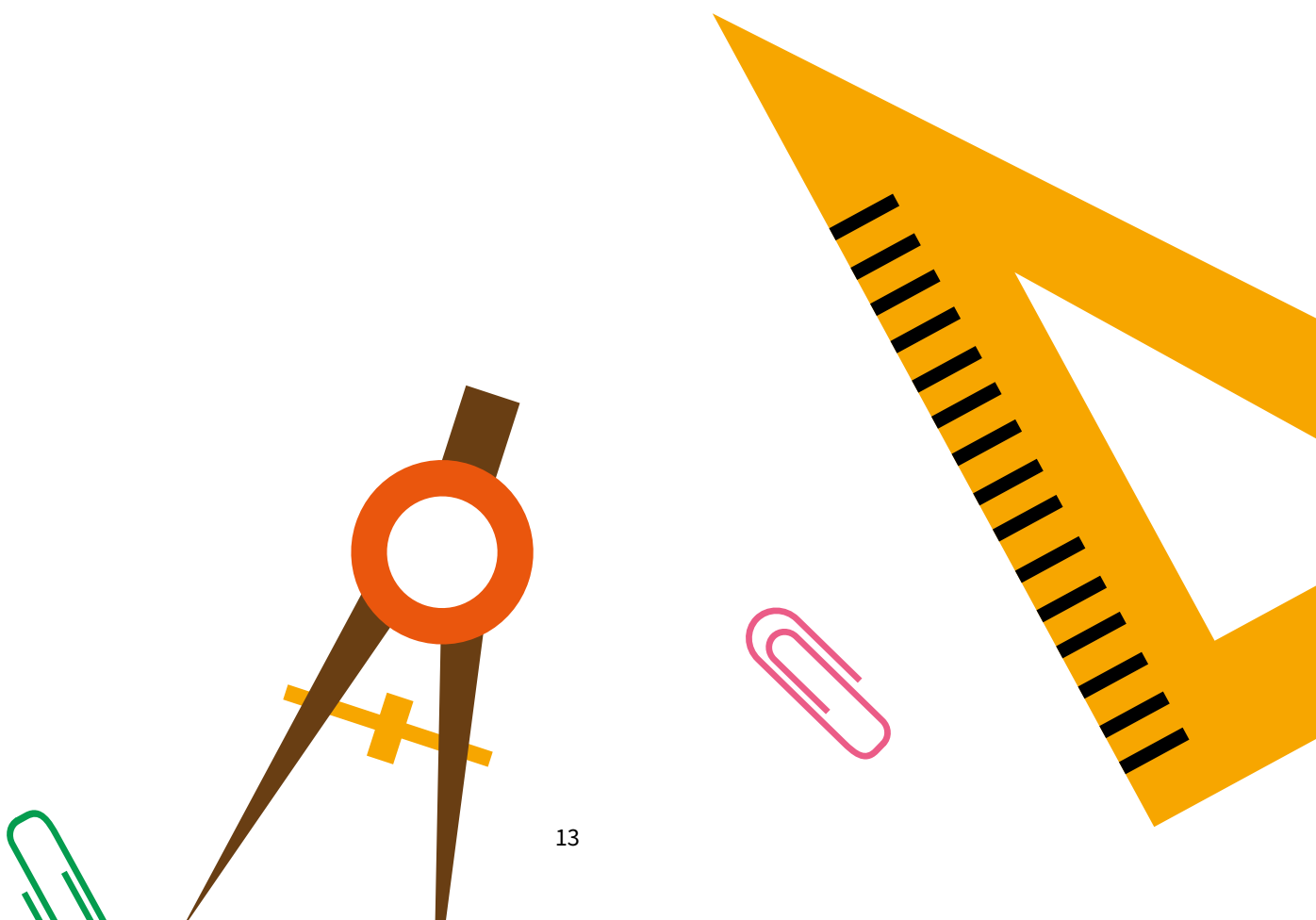
Programmeringen sker i programmeringsmiljön Scratch som körs i en webbläsare. Om det finns tillgång till internetförbindelse under verkstäderna behöver man inte ladda ner något på datorerna för programmeringen. Om det inte är möjligt att ordna internetförbindelse kan Scratch-programmet även installeras på datorerna i förväg. Läs mer om detta på sidan 20.

Inledning till matematik- uppgifterna

Kluraklubbs matematikuppgifter är problemlösningssuppgifter som utvecklar olika tankefärdigheter och samarbete. Syftet med dem är att erbjuda matematikstuderande i alla åldrar utmaningar och upplevelser av att lyckas. Matematik kan vara roligt och inspirerande och när man löser uppgifter får man även använda sin kreativitet.

I matematikuppgifterna i Kluraklubb ligger huvudvikten på att göra och förstå. I praktiken kan detta innebära att pyssla, spela, lösa problem och fundera. Uppgifterna kan lösas både ensam, i par och i små grupper. Problemen som löses tillsammans övar utöver problemlösningssjälvfärdigheter även upp samarbetsfärdigheter samt att ta andra i beaktande.

I handledningen av matematikavsnittet är det viktigaste överskådlighet och logik. En stor del av lösandet av problemen är självständig verksamhet för deltagarna. Det är bra att gå igenom reglerna för utmaningarna och spelen med exempel och till slut fråga om deltagarna har frågor. Det är bra att ge anvisningar om uppgifterna innan materialet delas ut till deltagarna så att koncentrationen inte störs.



Inledning till programmerings- uppgifterna

Programmeringsuppgifterna i Kluraklubb tränar basfärdigheterna i programmering och att använda datorer. Alla av de fem presenterade spelen lämpar sig även för nybörjare. I den använda miljön är det lätt att fortsätta öva sig på programmering självständigt efter verkstaden. I en del av anvisningarna presenteras även tilläggsfunktioner som kan tas med om tiden och deltagarnas kunskapsnivå tillåter detta. Programmering lärs nuförtiden ut även i skolor, så en del av deltagarna kan redan en del programmering från förut. Det är bra att utnyttja deras färdigheter då man löser de andra deltagarnas problem.

Bli inte skrämdd för att det här avsnittet är långt. Du behöver inte tillägna dig allt på en gång. Materialet blir så småningom bekant när du leder verkstäderna. Vid ordnandet av en verkstad räcker det att en av ledarna tar på sig rollen som ansvarig ledare för programmeringsavsnittet och går igenom dessa anvisningar. Då den ansvariga ledaren leder Kluraklubb-programmeringsavsnittet med sin kollega som hjälpledare, utvecklas hjälpledarens färdigheter samtidigt genom den praktiska verksamheten. Då hjälpledaren har fått självsäkerhet och erfarenhet av att leda kan hen handleda en ny hjälpledare vid följande verkstäder. Kompetensen ökar genom träning och barnen och de unga får njuta av verkstäderna genast från början.

PROGRAMMERINGSDELENS FRAMSKRIDANDE

I början av verkstaden är det bra att visa ett modellspel för deltagarna. Det finns en länk på webbplatsen <https://pulmaario.luma.fi/>. Handledaren presenterar sprajtarna i spelet och målet med spelet, varvid deltagarna får en allmän bild av verkstaden.

Handledningen av programmeringen av själva spelet sker stegvis. I varje skede visar först den ansvariga ledaren hur en ny del av ett spel är uppbyggd. Sedan går handledarna runt bland deltagarna och hjälper dem. Avsikten är att säkerställa att alla deltagare håller takten och har förstått tillvägagångssättet. Deltagarna kan ha väldigt olika färdigheter och därför varierar frågorna mycket. Ibland kan deltagarna handleda varandra.

I slutet av verkstaden är det bra att reservera lite tid för att testa spelet som nyss gjorts och presentera det för de andra deltagarna. Att presentera sina egna spel för de andra deltagarna är en viktig del av inläringserfarenheten: upplevelsen av att man själv klarar sig och kan ökar av kamratrespons.

Handledarens anvisningar är inget absolut rättesnöre och man måste inte bli färdig med spelet om tiden tar slut. Det väsentliga är att alla har något roligt att göra och utmaningarna inte blir övermäktiga. Det mest effektiva handledningssättet är att gå runt bland deltagarna och besvara frågor. Genom gemensam handledning strävar man efter att skapa en gemensam erfarenhet och rytm för arbetet. Målen för ett spelprojekt som görs under en verkstad ska därför hållas rimliga. Då handledaren för verkstaden och deltagarna är redo för det finns det även lämpliga tilläggsegenskaper för längre hunna i de verkstadsspecifika programmeringsanvisningarna.

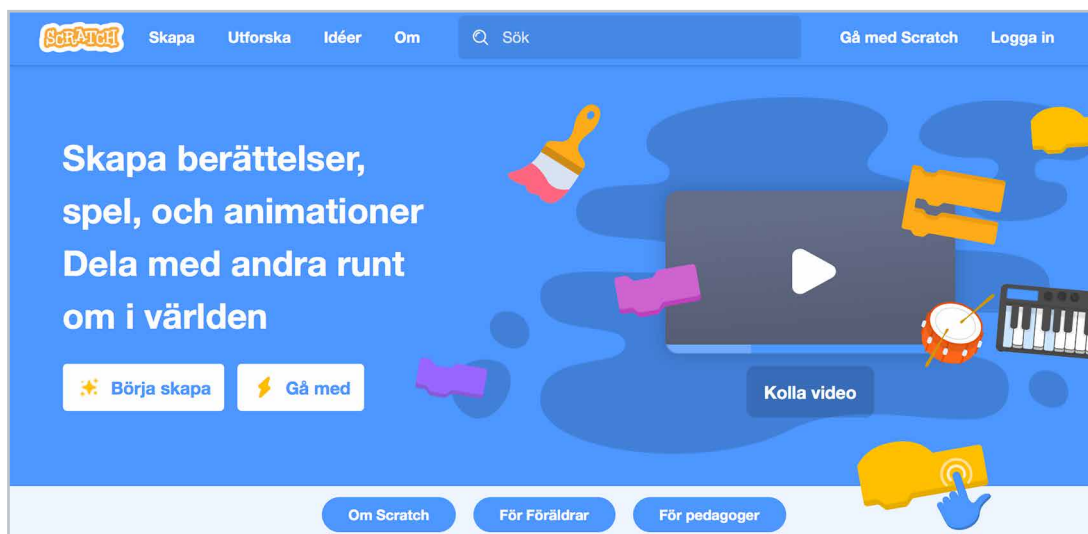
SCRATCH-PROGRAMMERINGSMILJÖN

Scratch är en webbaserad programmeringsmiljö, som riktas särskilt till barn och unga. Den är avgiftsfri och kräver inte inloggning. Scratch finns på adressen <http://scratch.mit.edu>. Samtidigt som du läser denna anvisning är det bra att öppna webbplatsen och testa saker som nämns i guiden så att miljön börjar kännas bekant.

Såsom även alla andra moderna program blir inte heller Scratch någonsin helt färdigt, utan det uppdateras kontinuerligt. Ofrånkomligt kommer även denna guide att i viss mån bli efter när det kommer nya egenskaper och idéer till Scratch. Grundtankarna och de vanligaste strukturerna inom programmering har emellertid inte ändrats under årtionden, så idéerna bakom anvisningarna förblir aktuella, även om små detaljer skulle ändras. Scratch är i alla fall ett utmärkt verktyg för att börja lära sig programmering.

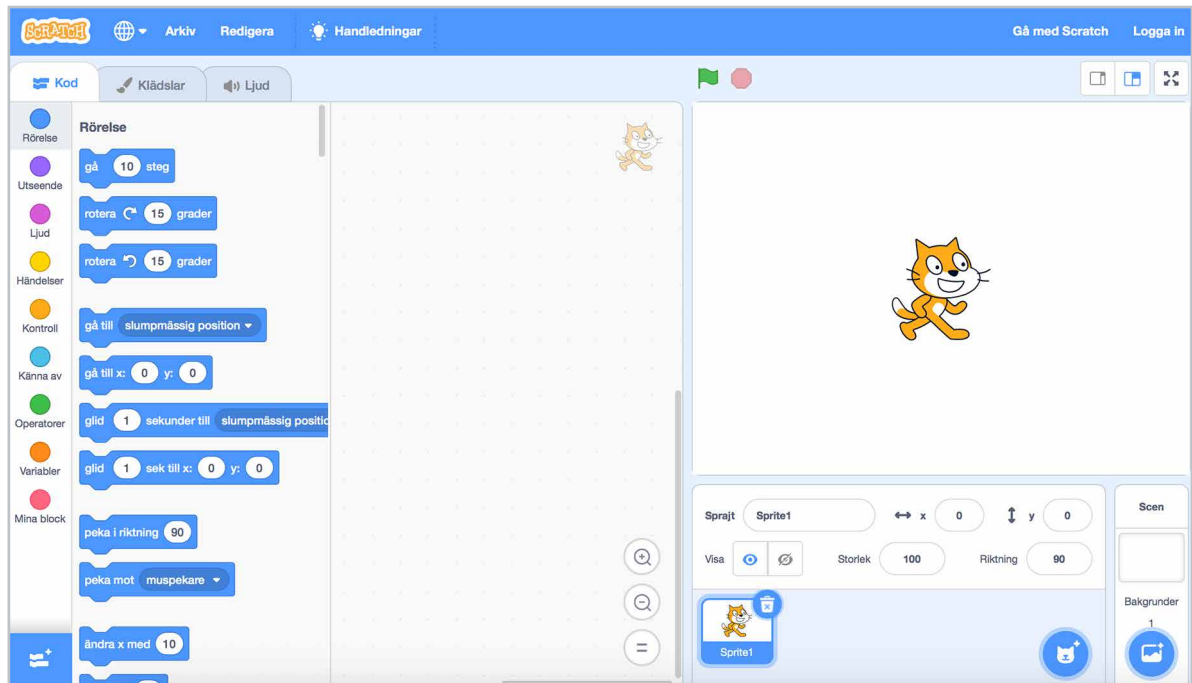
Scratch har översatts till flera språk, även till svenska. Språket kan väljas i menyn i den nedre kanten på förstasidan. Genom att klicka på "Gå med i Scratch" i sidans övre kant kan du skapa ett avgiftsfritt användarnamn. Webbplatsen sparar projekt som gjorts av inloggade användare automatiskt i Scratch-molnet. Om du inte skapar ett eget användarnamn kan du ändå ladda ned projektet på din egen dator. Vi rekommenderar emellertid att handledaren skapar ett användarnamn, till exempel med bibliotekets namn, för annars går allt gjort arbete förlorat till exempel vid en felsituation i webbläsaren. En inloggad användare kan även dela sina egna projekt.

Klicka först på "Skapa".



Programmeringsvy

Scratch programmeringsvy ser ut så här:



Om du glömde att välja språk på förstasidan kan du ändra språket även genom att klicka på jordglobsikonen i den blå balken upptill på sidan (på bilden bredvid ordet Scratch).

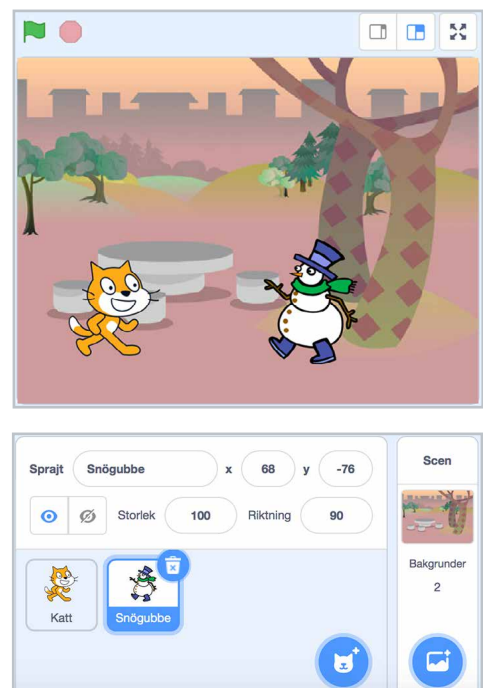
Testområde

Uppe till höger finns ett område avsett för testning av programmet. Programmet kan startas genom att klicka på den gröna flaggan och stoppas genom att klicka på den röda knappen.

Testområdet kan också förstöras till en "presentationsvy" så det täcker hela webbläsarfönstret genom att klicka på knappen uppe i högra hörnet.

Sprajtlista och bakgrunder

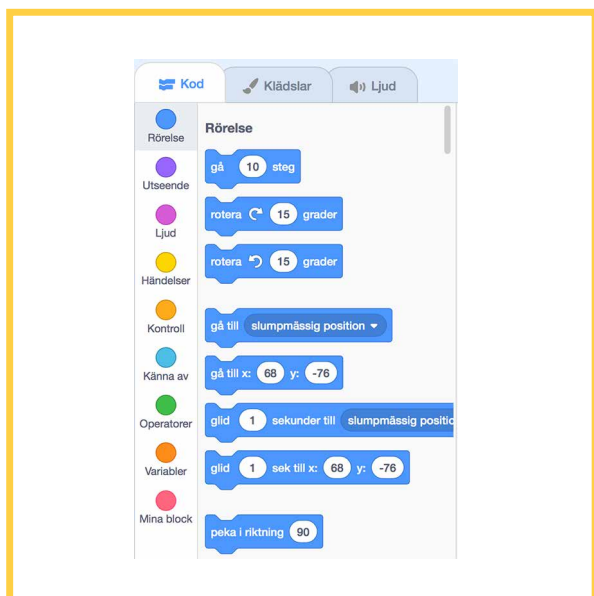
Sprajterna i programmet och bakgrunden har listats under testområdet. Det är möjligt att lägga till flera bakgrunder och sprajtar i programmet. **Nya sprajtar** läggs till genom att klicka på en kattikon i nedre högra hörnet av sprajtområdet. Genom att klicka på ikonen kan du välja från Scratch bildbank, som är det mest praktiska alternativet vid korta verkstäder. Genom att hålla musen över kattikonen kan du också se alternativ för att rita själv, välja en bild som finns på datorn eller ta en slumpmässig sprajt från Scratch bildbank.



Du kan också ändra **sprajtens namn**. Först klickar du på önskad sprajt varefter den omringas av en blå rand. Nu kan du se några av sprajtens egenskaper i området ovanför sprajtlistan. Där finns bland annat ett textfält med sprajtens namn som kan ändras.

Du kan också radera sprajtar med papperskorgikonen när sprajten är valt i listan, men då ska du vara försiktig. Då du raderar sprajten försvinner hela dess funktionalitet, det vill säga programmeringskoden. Deltagarna i verkstaden kan radera sprajten helt till exempel då de vill ändra sprajtens utseende. Om sprajten redan har en programmeringskod lönar det sig att istället för att radera sprajten, ändra dess klädsel, som du får läsa mer om lite senare. Om någon av deltagarna hinner radera sprajten av misstag går det eventuellt ännu att återställa den i menyn "Redigera" uppe på sidan, men det går bara att återta den senaste redigeringen.

Du kan lägga till **olika bakgrunder** på "Scenen" som finns till höger om sprajtlistan. Under den lilla bilden som föreställer projektets bakgrundsbild finns alternativ till nya bakgrunder som kan läggas till på samma sätt som det går att lägga till sprajtar. Genom att klicka på fotoikonen kan du bläddra mellan färdiga bakgrunder i Scratch bildbank.



Verktygslåda

Till vänster i programmeringsvyen finns Scratch verktygslåda. Ovanför lådan finns även flikar för de olika vyerna: *Kod*, *Klädslar* (eller *Bakgrunder*) samt *Ljud*. I de två sista kan du ändra sprajtarnas/ bakgrundernas utseende samt ljud. I Kluraklubb-guiden koncentrerar vi oss emellertid på programmering, det vill säga fliken *Kod* på vilken du hittar alla kommandon som Scratch förstår samt programmeringsområdet.

Användning av kommandon

Koden, det vill säga kommandona, har delats in enligt funktionssätt i **olika färger**. De spelkommandon som presenteras i denna guide kan hittas med hjälp av dessa samma färger.

I TEXTANVISNINGARNA FÖR VERKSTÄDERNAS SPEL NÄMNER VI FÖR ATT SPARA UTRYMME INTE FÄRGER, UTAN AVSikten ÄR ATT BILDERNA VID ANVISNINGARNA ANVÄNDS SOM TIPS FÖR ATT HITTA DE KOMMANDON SOM BEHÖVS.

När du hittar ett kommando som du vill använda, dra det till området höger om verktygslådan. Det här området är **programmeringsområdet**. Kommandona ges alltid till en viss sprajt, så säkerställ först att du valt rätt sprajt från sprajtlistan som presenterades ovan. Runt den valda sprajten finns en blå rand; även Scenen kan

vara vald. Obs! Scenen kan bland annat inte röra sig så den saknar många sådana kommandon som sprajtarna har.

Kommandona kan kopplas ihop som byggklossarna i en byggsats. Varje block kopplas ihop med de andra blocken i det övre vänstra hörnet. Ofta börjar kommandoserien med det gula blocket Händelser. Med hjälp av det vet programmet när kommandoserien i fråga ska utföras. Då händelsen i det gula blocket har skett börjar programmet utföra kommandona som fästs i blocket i ordningsföljd uppifrån och ner. En del av kommandona kan innehålla **upprepningar** eller **villkor** som ändrar ordningen på utförandet. Dessa tas upp senare i samband med spelreglerna i guiden.

De kommandon som inte behövs kan raderas genom att dra dem tillbaka till verktygslådan. En kommandoserie som har raderats av misstag kan återställas genom att högerklicka på programmeringsområdet och välja "ångrä". Det är bäst att göra detta genast efter misstaget eftersom möjligheten att ångra försvinner bl.a. när man klickar på en annan sprajt.

Ritande och ljud

Många unga programmerare tycker mycket om att rita egna sprajtar. Ljud ger spelprojekten helt nytt liv. I Kluraklubb finns det emellertid ofta inte tid för dessa och gränserna för utrustningen som används kan komma emot. Därför gås dessa teman inte igenom i denna guide. Om någon deltagare är så snabb att hen ritar sina egna sprajtar under verkstaden eller hinner lägga till ljud i spelet är det bra att beakta några saker.

Bilderna på sprajtarna måste alltid centreras. Detta sker i ritvyn (dvs. fliken Klädslar). Hela sprajten ska dras så nära ritområdets mittpunkt som möjligt, med hjälp av ett runt märke som finns där. Om det finns



PÅ HANDLEDARENS DATOR
LÖNAR DET SIG ATT FÖRSTORA
PROGRAMMERINGSOMRÅDET
SÅ ATT DE VALDA BLOCKEN SYNS
BÄTTRE PÅ TAVLAN.

OVANFÖR TESTOMRÅDET
FINNS DET TVÅ IKONER SOM
PÅVERKAR DE RELATIVA
STORLEKAR AV DE OLIKA
OMRÅDEN I
PROGRAMMERINGSVYN.

DET ÄR OCKSÅ BRA ATT FÖRSTORA
BLOCKENS STORLEK. DETTA SKER
GENOM ATT KLICKA PÅ +-KNAPPEN
I KOD-FLIKEN SOM FINNS I DET
NEDRE HÖGRA HÖRNET I DET STORA
PROGRAMMERINGSOMRÅDET. DU
MÅSTE DÖLJA INSTRUKTIONSRUTAN
SOM IBLAND DYKER UPP PÅ SCRATCH
PROGRAMMERINGSOMRÅDE FÖR ATT
DU SKA HITTA KNAPPEN.

OBS! ZOOMNINGEN FUNGERAR
BARA FÖR BLOCKEN PÅ
PROGRAMMERINGSOMRÅDET,
BLOCKLISTAN I VERKTYGSLÅDAN
FÖRSTORAS INTE.

funktionsfel i en deltagares spel som känns oförklarliga är orsaken mycket ofta att sprajten inte har centererats. Ofta måste man efter centreringen justera blocken "gå till punkten", "glid till punkten" eller "sätt x/y till" som används i programmet på nytt.

Scratch har två olika ritsätt, ett bitmappsläge i stil med Paint samt ett vektorläge. Även om sprajtar som ritats i vektorläget tål till exempel en ändring av storleken utan att bli korniga uppstår det ibland problem i användningen av dem. Ibland fungerar inte kontrollen av hur vektorformade sprajtar rör varandra. Då hjälper det ofta att ändra den färdiga bilden till bitmap genom att klicka på knappen nedanför ritområdet.

Storleken på sprajten kan ändras i sprajtens egenskaper eller genom att programmeras. I ritvyn lönar det sig i allmänhet inte att ändra storleken på klädseln utan att bildens upplösning försämras. Några av sprajtens egenskaper kan vara gömda om programmeringsområdet är förstorat (se tips på förra sidan). Som standard är storleken 100, alltså går det bra att tänka på storleken som procenthalten.

Utöver bildbiblioteket finns det i Scratch ett omfattande ljudbibliotek med bland annat ljudeffekter, skalor för olika instrument och färdiga loopar. Det går också att ladda upp ljudfiler från datorn och banda egna ljud om det finns en mikrofon i datorn. Observera dock att det inte nödvändigtvis går att spela eller banda ljud eller överföra filer om du öppnat ett inkognitofönster i webbläsaren. Testa därför på förhand med utrustningen som används om ljudbehandling är möjligt om du vill ha med ljud i verkstaden. Ljud behandlas i programkoden med kommandona i den lila fliken Ljud.

Reservplaner

Scratch-webbplatsen kan slumpmässigt vara otillgänglig på grund av uppdateringar, fel i internetförbindelsen eller andra orsaker. Du kan förbereda dig på det här på förhand genom att ladda versionen "offline-redigerare" av Scratch i punkten **Download** längst ner på Scratch-hemsidan. Det finns flera olika versioner av Offline-redigeraren. Om det är möjligt lönar det sig att installera **version 3.0** eller en nyare. Scratches yttre har förändrats mycket sedan version 2.0, och exempelvis har version 1.4 inga kloner eller block vilket du bör beakta för programmeringsuppgifterna 1, 4 och 5.

Ett annat alternativ är att testa en miljö som heter Snap! (<http://snap.berkeley.edu/>) som påminner om Scratch men finns på en lite annorlunda plattform. Snap! fungerar emellertid på vissa punkter på ett lite annat sätt så vi rekommenderar att ni testat Snap! endast om handledaren har haft tid att själv på förhand göra ett spel i Snap!-miljön.

HANLEDNING AV PROGRAMMERINGEN

Den ansvariga handledarens roll

Den ansvariga ledaren har gjort verkstadens spel på förhand helt själv. Hen har också funderat på hur hen på bästa sätt organiserar handledningen på ett för hen själv naturligt sätt samt funderat på punkter som är problematiska för hen själv och att beskriva dem. Det är bra att först testa att handleda en testpublik, till exempel verkstadens andra handledare. Handledandet innebär i praktiken att förklara sin egen programmering så enkelt som möjligt, med hjälp av vilket man eftersträvar att förklara programmerarens tankemodell. Det är emellertid möjligt att ha med guiden under verkstaden uppslagen på rätt ställe.

Det enklaste sättet att växa till ansvarig handledare är att först vara hjälpledare, det vill säga i rollen av en "gesäll" och lära sig under en verkstad då någon som redan tidigare har satt sig in i ledandet av verkstaden är den ansvariga handledaren. Det är emellertid lätt att lära sig hur programmeringsmiljön fungerar och du behöver inte vara rädd för att göra fel. Därför ska du testa projektet hela tiden. Syftet med testandet är att rätta till både tankefel och små misstag.

Då du står framför gruppen lönar det sig att fästa uppmärksamhet vid åskådligheten. Det är särskilt svårt för barn och ungdomar att se små rörelser med en mus som projiceras på en stor vägg. Om du pekar på rätt punkt med fingret gör det lättare att följa.

Handledningen som sker framför gruppen och innebär att i ord beskriva programmeringskoden är endast en del av tiden i verkstaden. Resten av tiden följer den ansvariga handledaren liksom de andra handledarna att deltagarnas spelbyggande framskrider och svarar på frågor som dyker upp.

Den assisterande handledarens roll

Det är ofta mest praktiskt att en handledare som assisterar den ansvariga handledaren använder en dator som är kopplad till presentationsutrustningen då den ansvariga handledaren förklarar nästa programmeringsskede framför gruppen. Handledaren vid datorn ska inte förutspå programmeringen. Hen lyssnar till huvudhandledaren och programmerar efter det. På så sätt kommer eventuella oklara anvisningar snabbt fram.

Handledaren vid datorn ska inte sträva efter att göra det som sägs så snabbt som möjligt utan det får ta tid att dra blocken. Alla som följer handledningen ska hinna se varifrån blocket kommer och vart det förs.

När det är paus i den aktiva handledningen då deltagarna gör de arbetsskeden som de just fått höra om, lämnar den assisterande handledaren datorn och hjälper deltagarna på samma sätt som den ansvariga handledaren. Det lönar sig inte att lägga ner för mycket tid på att finslipa utseendet på spelet som handledarna gör, eftersom det ofta är många som behöver hjälp.

Tips för verbalisering av programmeringen

Programmering är ytligt sett bara organisering av mekaniska kommandon men samtidigt planerar programmeraren programmets struktur samt hanterar funktionerna av olika delar i programmet i relation till varandra samt användaren. Vid verbalisering av programmering är det fråga om att överföra dessa färdigheter och tankemodeller till åhörarna. För att klä olika saker i ord behövs lite olika sätt att närma sig. Dessa kan tas med i den egna handledningen allt efter som de känns naturliga. Nedan finns några tips, mer hjälp finns på webbplatsen <https://pulmaario.luma.fi/>.

Färger, placering och former

I Scratch-programmeringsmiljön är de olika funktionerna färgkodade och de finns alltid under samma block. **Det lönar sig att utnyttja färgerna och placeringarna** i handledningen. Programmeringsblockens **former** ger även tips på hur de används.

Handledaren kan beskriva den första flytten av programmeringsblocken till programmeringsområdet till exempel så här:

*“Vi börjar med att gå till den gula menyn **Händelser**. Den finns i mitten av verktygslådan.”*

[samtidigt pekar den ansvariga handledaren på rätt punkt på tavlan; den andra handledaren klickar på detta ställe]

*”I Händelser finns block med vilka man börjar med skript. Dem känner man igen på den **runda överkanten**. Dem kan man inte fästa i den nedre kanten av andra block, utan de kommer alltid i början av ett skript.*

*Dra från **Händelser** det **översta** gula blocket där det står ‘när den gröna flaggan klickas på’..”*

[samtidigt pekar den ansvariga handledaren på rätt block på tavlan; handledaren vid datorn drar blocket till programmeringsområdet]

Skriptens struktur kan kommenteras till exempel så här:

*”Figuren rör sig **hela tiden**. Vi skulle kunna upprepa samma kommando många gånger, men det blir mycket arbete. Datorer är bra på att upprepa saker automatiskt, därför använder vi den färdiga **repetera-strukturen**. Vi vill att sprajten rör sig **under hela spelet** så vi väljer det ljusorange **för alltid** -blocket. Blocket har två **förgreningar**. In i det sätter vi de kommandon som vi vill upprepa.”*

Målet är att deltagarna också om de vill kan programmera utifrån de verbala anvisningarna. En del av ungdomarna gör sitt eget program samtidigt då du undervisar framför gruppen och kastar bara sporadiskt en blick på tavlan. En del tar modell av det som finns på tavlan och gör genast detsamma. Andra vill däremot först se instruktionerna och därefter själva göra det som de ser på tavlan. Du kan stödja de olika sätten genom att i tillräckligt långsam takt ge instruktioner så de lämpar sig för de olika sätten.

Programmeringsprojektets struktur

I handledningen är det bra att även vid lämpliga tillfällen tala om hur projekthelheten formas. När ni inleder ett nytt skript kan du konstatera att för att programmeringskoden ska vara tydlig, strävar ni efter att **varje skript ansvarar för bara en sak**. Till exempel då ni övergår till att behandla sprajtens reaktioner till en annan sprajt börjar ni ett nytt skript.

Det är även viktigt att se till att rätt skript kopplas till rätt sprajtar, särskilt eftersom avsikten är att spelets sprajtar reagerar på varandra. Att programmeringskoden är planmässig är nödvändigtvis inget villkor för att programmet ska fungera men det är väldigt viktigt för att förstå programmet. Programmeringskoden skrivs i det här tänkesättet egentligen inte för datorerna utan för de andra programmerarna.

Anvisningarna för verkstädernas spel i denna guide har planerats färdigt, så verkstadens handledare behöver inte själva planera strukturerna. Därför kan det i det här skedet kännas otroligt svårt att tala om programmets struktur. Det viktigaste är emellertid att ta tjuren vid hornen och börja ordna verkstäder även om verbaliseringen av din egen programmering främst grundar sig på att beskriva hur du mekaniskt flyttar programmeringsblocken. Det räcker mer än väl för att ordna dessa verkstäder. Då du lär dig mer kan du ta med element som beskriver logiken i programmeringskoden.

Programmering i par

Det ska finnas minst en dator och mus per två deltagare. Naturligtvis är det fint om du kan erbjuda varje deltagare en egen dator men det kan även vara till stor nytta att programmera i par då man tillsammans kan fundera på saker.

Programmering i par används i stor utsträckning inom professionell programmering samt som en pedagogisk metod inom undervisning. För programmerarna är kärnan i att programmera i par att den ena personen funderar på vilka funktioner som behövs och spelets mer omfattande linjer då den andra igen svarar för programmeringen. Personen som sitter bredvid är ingen passiv åskådare eller elev utan hen ska aktivt delta i det som sker till exempel genom att lägga märke till fel, söka efter kommandon och föreslå test. Hen kan till exempel vara den person i paret som noggrannare följer instruktionerna och ser till att alla skeden blir gjorda.

Programmering i par kräver emellertid även särskild uppmärksamhet av handledarna så att **bägge parter får programmera ungefär lika mycket**. I verkstaden lär deltagarna sig även att använda datorer, så ledarna måste se till att den skickligare parten inte hela tiden håller i musen. Annars drar sig den mer oerfarna parten lätt ännu mer undan även om hen skulle vilja göra själv.

Den ansvariga ledaren kan säga till exempel var femte minut att **den som håller i musen ska ge den till sitt par**. Det är bra att handledaren känner efter vilken rytm som passar för handledningen. Bytet kan ske till exempel alltid då man går över till nästa skede i handledningen. Om du följer anvisningarna i guiden noggrant punkt för punkt, till exempel när du kommer till en ny numrerad punkt i anvisningarna.

En del av deltagarna kan redan ha erfarenhet av att använda Scratch då verkstaden inleds. Det är bra att para ihop de mest erfarna deltagarna och också ge dem förslag om tilläggsuppgifter. Dock kan det i gruppen också finnas naturliga handledarförmågor som blir ivriga att ge programmeringsråd till sin mindre erfarna kompis. Vanligtvis har barn och unga emellertid inte tålamod till detta så det är bättre om de mer erfarna sparrar varandra och nybörjarna får fundera i lugn och ro tillsammans.

Svåra frågor av deltagarna?

Du behöver inte vara rädd för att bli tvungen att säga till en deltagare att du inte vet hur något som inte hör till verkstaden görs. Det lönar sig att uppmuntra deltagaren att hitta svar på frågorna genom att de själva testar. Så uppmuntrar du till egen aktivitet då deltagarna ska lära sig att använda programmeringsutrustningen.

Ibland räcker det emellertid inte med att testa. Då går det att ställa frågor till **vetenskapsklassen Linkkis Spelklíník-studio** som du hittar genom att söka efter "peliklinikka" i Scratch-miljön. Trots det finska namnet får du hjälp även på svenska. För att fråga behöver du ett eget Scratch-konto. Handledarna av verkstaden kan vid behov ställa frågor i deltagarnas ställe till Spelkliniken. Ni kan till exempel reda ut saker inför nästa verkstad via denna.

Härnäst presenteras uppgifterna för Kluraklubbs fem första verkstäder. Lycka till med handledningen av Kluraklubb-verkstäderna!



Anvisningar för verkstäderna

?

?

?



?



?

?

?

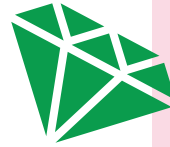
?

?

?



FÖRSTA VERKSTADEN



Juveltjuven

Ni är listiga juveltjuvar som har hört att stadens rika hertiginna har satt in en stor mängd vackert gröna ädelstenar på den lokala banken. Ni planerar ett bankrån men ni har ännu ingen lämplig säck för att transportera ädelstenarna. Bygg en så stor säck som möjligt av de block som ni har. Säcken måste vara slitstark och den får inte ha några hål så att ädelstenarna inte faller ner på golvet. Hurdan form borde säcken ha för att rymma mest ädelstenar? Klarar ni av själva rånet eller fastnar ni i laserfällan?

Planerande av ädelstenssäcken är verkstadens matematiska problem som heter Volymutmaningen. Uppgiften är att göra en så stor säck som möjligt av de givna materialen. Det ger övning i att uppfatta volym och samarbetsfärdigheter. Själva rånet sker genom programmering. Tjuven syns i mitten av spelområdet och kastar spegeldamm på rörelsedetektorernas laserstrålar så att hen inte upptäcks.



LITTERATURTIPS

- Dan Höjer: **Circusdeckarna och juvelmysteriet**
- Martin Widmark: **Diamantmysteriet**
- R. F. Symes & R. R. Harding: **Kristaller & ädla stenar Oikukkaan diivan tapaus**



VOLYMUTMANINGEN

Tillbehör

- 1 påse (500 g) torkade ärter
- färgat utskriftspapper
- 1 rulle genomskinlig tejp
- sax
- 2 st. genomskinliga engångsmuggar
- permanent tuschpenna
- en pappkartong eller annan behållare för ärterna
- papper på volymutmaningen (som bilaga på sidan 81)

Förberedelser

Kopiera det bifogade volymutmaningspappret på färggrant utskriftspapper. Klipp pappret itu längs den streckade linjen. Volymutmaningen görs i grupper på 3–4 personer. Det behövs ett halvt volymutmaningspapper per grupp. Till exempel i en verkstad med 20 deltagare där man delar in sig i fem eller sex grupper behövs tre kopior (dvs. 5–6 halvor) av utmaningsvolympappret.

Häll ut ärterna i en papplåda eller annat större kärl. Vid mätningen fylls resultaten med ärter. Det är bra att fylla på ärter ovanför det stora kärlet så att det inte faller ärter på golvet.

Uppgiftsanvisningar

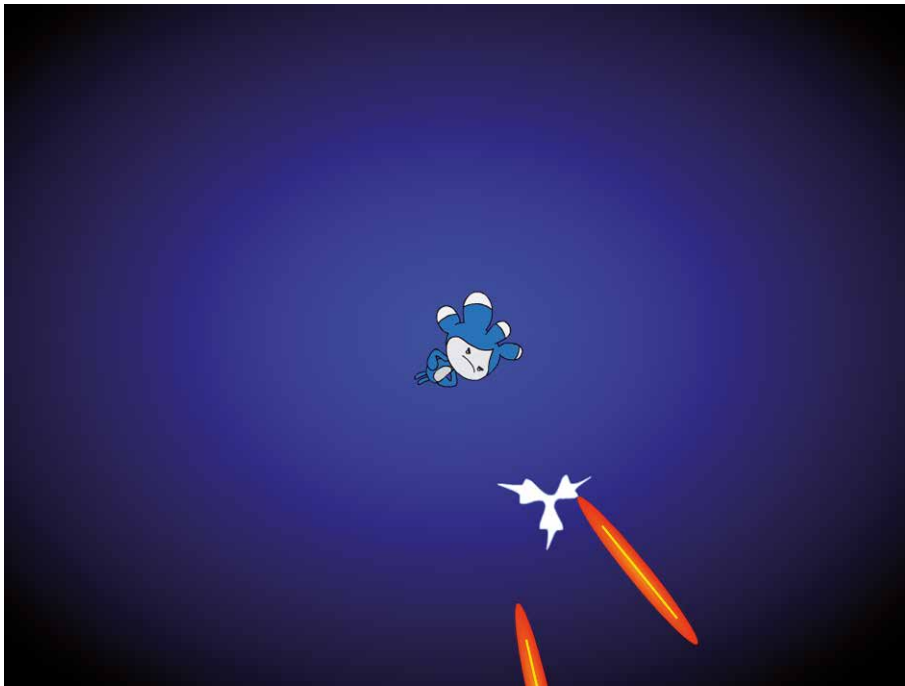
Byggskedet

1. Dela in deltagarna i grupper på 3–4 personer.
2. Ge varje grupp en halva av volymutmaningspappret, ungefär en meter tejp samt en sax. Den ena tejpändan kan fästas till exempel i ett bordshörn och den andra i golvet. Grupperna kan själva klippa mindre bitar av tejp.
3. Säg till grupperna att klippa loss sex bitar från volymutmaningspappret. Då borde varje grupp ha 2 st. A-kvadrater, 2 st. B-triangelar samt 2 st. C-rektanglar.
4. Grupperna har i uppgift att av det givna materialet bygga ett kärl vars volym är så stor som möjligt. Deltagarna får om de så vill vika eller klippa pappersbitarna i mindre bitar och använda sin fantasi då de använder tejp. Det lönar sig emellertid för deltagarna att ta i beaktande att de bara har en begränsad mängd tejp till sitt förfogande och att kärlet måste vara starkt och inte får ha några hål. Efter byggskedet fylls kärlen med ärter och ärternas volym mäts.
5. Då du gett instruktionerna kan byggandet börja. Byggtiden är 15 minuter.

Mätning

6. När tiden är slut ska kärlen vara färdiga. Be grupperna att i tur och ordning komma till mätplatsen.
7. Grupperna fyller sina kärl med ärter till brädden men ändå så att ärterna hålls inuti kärlet.
8. Ärterna hålls därefter försiktigt över i plastglas. Anteckna hur många glas som fylldes till brädden av ärter samt märk ut med permanent tusch med ett litet streck på det glaset som inte blev helt fullt hur fullt det blev och vilken grups resultat det var fråga om.
9. När alla grupperns kärl har mätts, tillkännage resultaten och titta tillsammans vilken grupp som hade det rymligaste kärlet. Be även grupperna att berätta för alla hurdana idéer de andra grupperna hade.

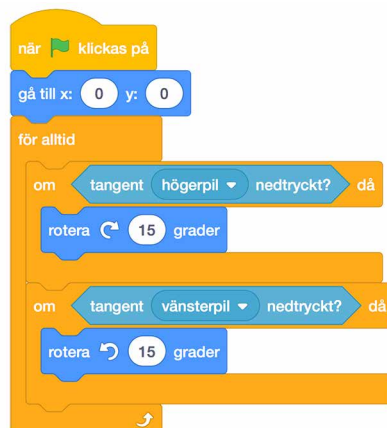
FALLET MED RUBINEN



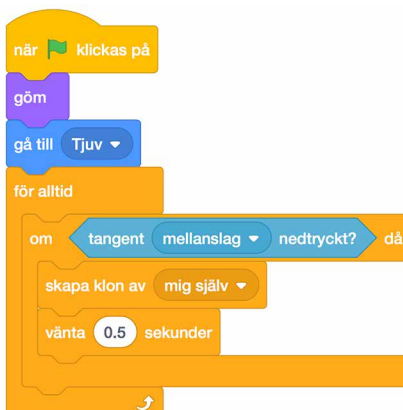
Tjuven är mitt i rutan och rör sig inte bort därifrån. Från fältets kanter närmar sig rörelsedetektorernas laserstrålar. Du måste kasta spegeldamm på laserstrålarna så att tjuven inte tas fast under rånet. Hur länge kan spelaren dra ut på sitt plundringståg utan att tas på bar gärning?

1. Skapa en ny sprajt: Tjuv och namnge den (se. s. 17).

2. Programmera sprajten att vända sig genom att man använder piltangenterna under hela spelet. Sprajten börjar mitt i rutan i origo, det vill säga i punkten $x = 0, y = 0$.

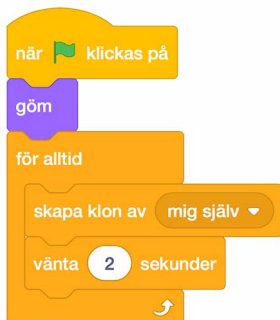


3. Skapa en ny sprajt: Spegeldamm och namnge den. Göm sprajtens "spegeldammslager" (dvs. den ursprungliga sprajten) och skapa där spegeldamm (dvs. kloner av sprajten) alltid då spelaren trycker på mellanslag. Ställ även in en begränsning för kasthastigheten för spelaren (kommandot vänta). Även denna process repeteras under hela spelets gång.



4. Efter att ha uppstått vänder sig spegeldammsklonen genast så att den pekar i samma riktning som tjuven. Sedan blir klonen synlig och börjar röra sig tills den når kanten, då den raderas. Det räcker inte med att gömma klonen på nytt för att få den att försvinna, eftersom Scratchs motor inte orkar göra ett gränslöst antal kloner. De skulle slutligen ta slut.

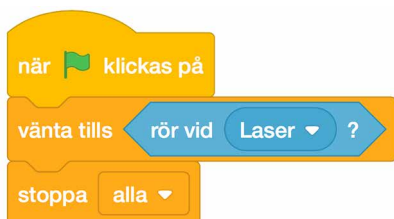




5. **Skapa en ny sprajt: Laser och namnge den.** Skapa laserstrålar från början av spelet med 1–2 sekunders intervall. Göm ”källan” till laserstrålen såsom spegeldammet och skapa kloner av den som vi snart berättar mer om.

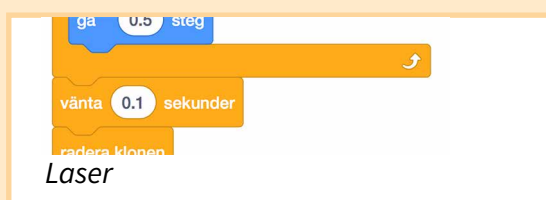


6. Avsikten är att laserstrålarna ska synas från spelområdets kanter. Det enklaste sättet är att först föra laserklonen till origo. Sedan väljer klonen en slumpmässig position tills den rör vid kanten. Eftersom klonen fortfarande är osynlig kan spelaren inte se det hända. Först därefter svänger laserklonen mot tjuven, blir synlig och börjar röra sig långsamt framåt. Rörelsen avbryts om den träffar spegeldamm. Efter det här raderas klonen.



7. Gå till tjuvsprajts skript. Programmera en ny kommandoserie för tjuven där den känner efter om laserstrålen träffar. Om laserstrålen träffar blir fallet med rubinen kort och spelet avbryts.

8. Valfritt: även spegeldammet kan försvinna när det träffas av en laserstråle. Då måste både spegeldammet och laserstrålen vänta en stund innan de försvinner så att båda hinner märka att de krockat. Lägg alltså till kommandot vänta till laserstrålens rörelse-kommandoserie. Därefter ska du lägga till en helt ny kommandoserie till spegeldammet där klonen väntar tills den träffar laserstrålen och försvinner därefter efter att ha väntat en stund.







ANDRA VERKSTADEN

Havsäventyret

Ditt lugna havsäventyr får en spännande vändning då flaggan på ett fiendefartyg syns vid horisonten. Fienden är beredd att sänka alla fartyg i din flotta. Hinner du sänka motståndarens fartyg innan hen sänker dig? Samtidigt samlar din sällskapsfisk in skatter som sjunkit till botten. En hur stor skatt hinner den samla in innan den lokala havsväktaren slutgiltigt skrämmer bort den?

Verkstadens tema är koordinatsystemet. I matematikdelen bekantar ni er med koordinatsystemet genom det traditionella spelet Sänka skepp. I programmeringsdelen gör ni ett maritimt insamlingsspel där sällskapsfisken flyttas genom att ändra dess koordinater.



LITTERATURTIPS

- Irene Trimble:
Pirates of the Caribbean: död man's kista
- Frida Nilsson: **Ishavspirater**
- Aleksy Bardy: **Oskars båtbok**



SÄNKA SKEPP

Tillbehör

- pennor
- spelplan för sänka skepp (som bilaga på sidan 82)

Förberedelser

Kopiera bilagan med spelplanerna för Sänka skepp till alla deltagare.
På spelplanen finns både ditt och kompisens koordinatsystem.

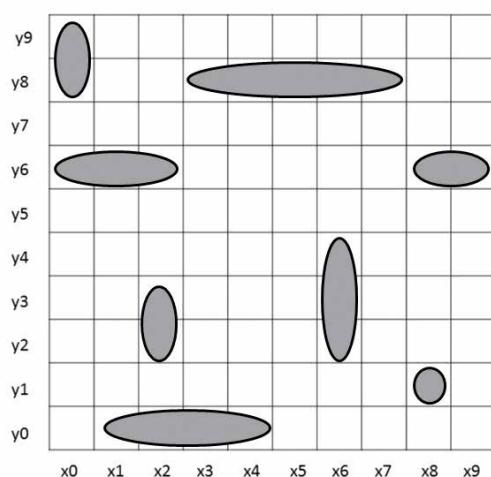
Uppgiftsanvisningar

Spelet spelas i par. Varje spelare behöver en spelplan och penna.

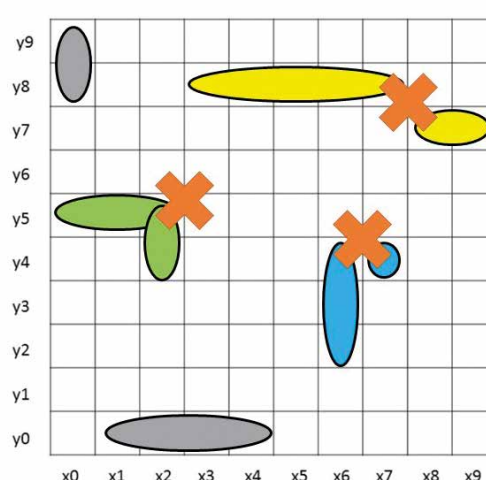
Att märka ut de egna fartygen i koordinatsystemet

1. Innan ni börjar spela märker varje spelare ut sitt fartyg i koordinatsystemet. I grundversionen kan fartygen både vara vågrätt eller lodrätt och de får inte röra i varandra. Reglerna kan ändras efter deltagarnas önskemål, men det är bra att förklara reglerna för alla genom exempel innan de börjar rita fartygen.
2. Sammanlagt åtta fartyg ritas:
 - 1 hangarfartyg (fem rutor långt)
 - 1 slagskepp (fyra rutor långt)
 - 2 kryssare (tre rutor långa)
 - 3 jagare (två rutor långa)
 - 1 u-båt (en ruta)
3. Då båtarna ritats kan spelet börja!

Ditt koordinatsystem



Kompisens koordinatsystem

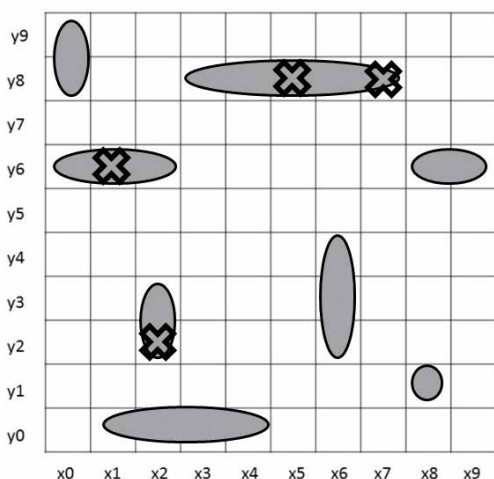


I bilden till vänster har fartygen placerats rätt. I bilden till höger har en del av fartygen placerats fel: de gröna fartygen har placerats på varandra, de gula fartygen snuddar vid varandra diagonalt och de blå fartygen är bredvid varandra och tar i varandra.

Spelets gång

4. Den yngre spelaren börjar. Spelaren väljer i kompisens koordinatsystem en ruta som hen vill skjuta på. Spelaren säger koordinaterna för den här rutan, till exempel x2, y5.
5. Om något av kompisens fartyg finns i rutan som spelaren nämner, säger kompisens "träff" och spelaren sätter ett kryss i rutan i fråga i kompisens koordinatsystem. Även kompisens sätter ett kryss i sitt eget koordinatsystem i rutan som fått en träff.
6. Om det inte finns ett fartyg i rutan sätts ett kryss i kompisens koordinatsystem. Om det blir träff får spelaren fortsätta annars går turen över till kompisens.
7. Om ett skott träffar varje ruta för ett fartyg sjunker fartyget. Spelaren ska berätta om att fartyget sjunkit. Det lönar sig för spelaren att märka ut ett fartyg som sjunkit i kompisens koordinatsystem genom att ringa in kryssen som står för det, eftersom det inte kan finnas ett fartyg bredvid det.
8. Spelets vinnare är den som först har lyckats sänka alla kompisens fartyg.

Ditt koordinatsystem



Kompisens koordinatsystem

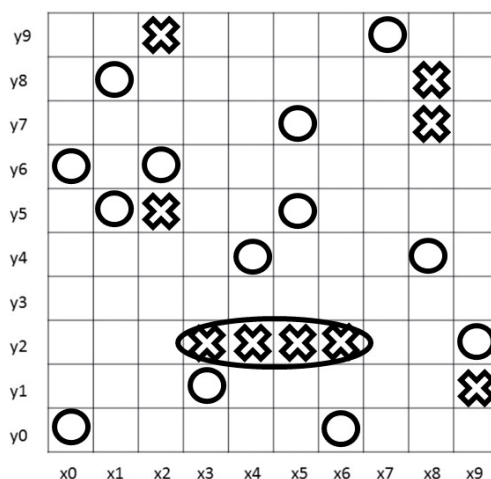
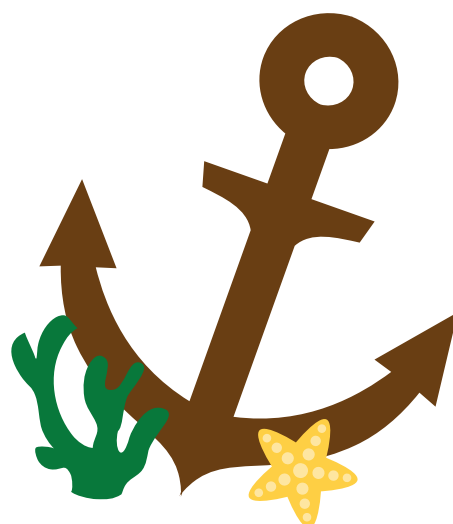
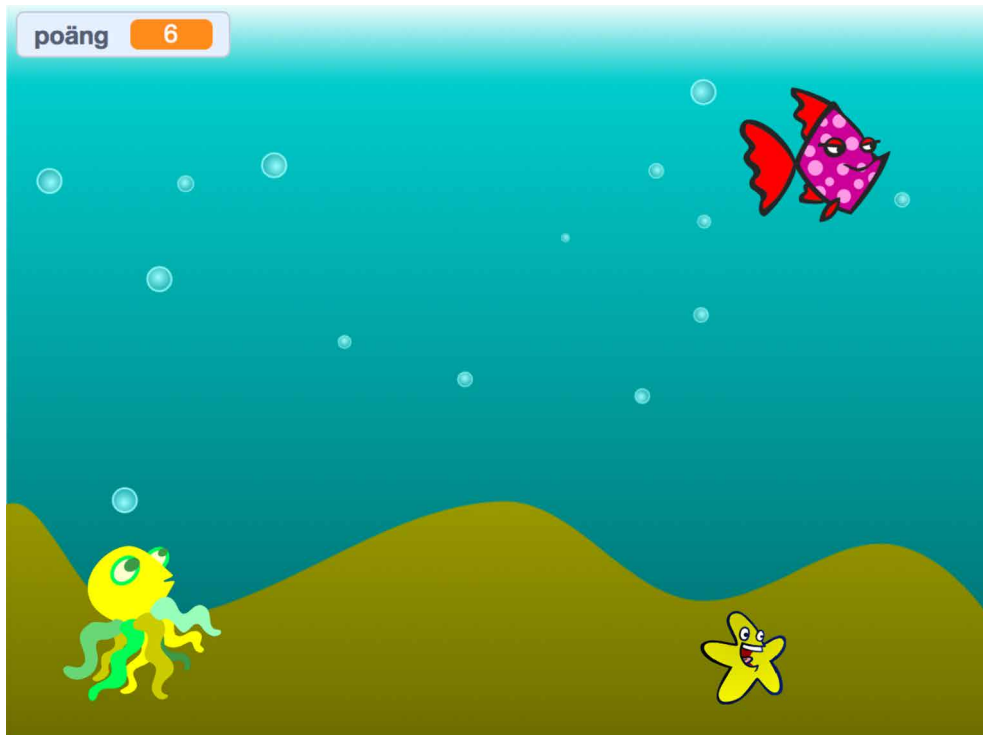


Bild på spelarens spelplan: I kompisens koordinatsystem har märkts ut med kryss de fartyg som fått en träff och med en cirkel de skott som gått förbi. En av kompisens fartyg har sänkts och det har ringats in. I det egna koordinatsystemet har märkts ut med kryss de skott som träffat de egna fartygen.



HAVSINSAMLINGEN

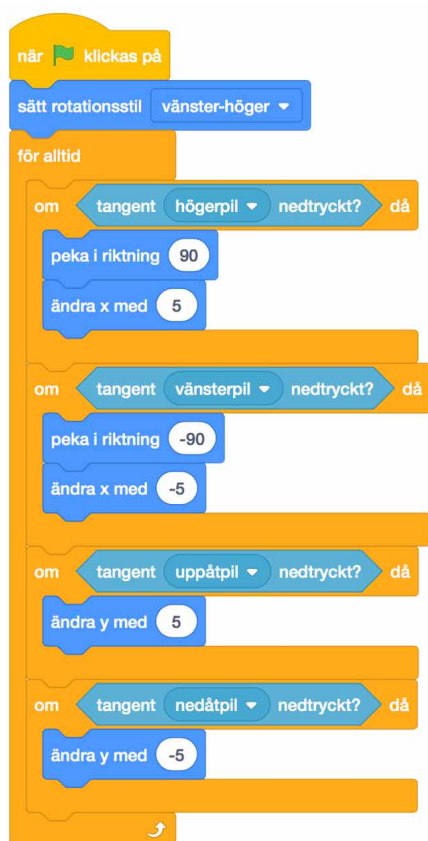


I spelet samlar man in skatter på havsbotten. Spelaren styr en av sprajtarna med piltangenterna. I rutan dyker på en slumpmässig plats upp en skatt som ska samlas in för att man ska få poäng. I spelet finns också en valfri fiende som försöker få fast spelarens sprajt.

1. Skapa en ny sprajt: **Spelare**. Namnge sprajten (se s. 17).

2. Spelaren rör sig **för alltid** med piltangenterna. För att röra sig används koordinatsystemet, det vill säga då man trycker på knappen **ändras spelarens x- och y-läge**. Modellkoden finns lite längre ner. Den ser invecklad ut men i den har man i själva verket bara upprepat liknande kommandon för de fyra piltangenternas riktningar.

2.1. Finjustering: om sprajten har en tydlig riktning i vilken dess näsa pekar, lönar det sig att även ändra riktningen i vilken sprajten pekar när du vänder den åt vänster eller höger. Dessutom kan du i början sätta till en rotationsstil.



3. Skapa en ny sprajt: **Skatt** och namnge den.

4. Skatten dyker i början av spelet upp på ett slumpmässigt ställe. Den väntar där tills spelaren får fast den. Därefter hoppar den till ett nytt ställe.

Programkodmallen finns under följande punkt.

5. **Skapa "variabelpoäng" åt alla sprajter.** Variabeln skapas i den mörkorange menyn Variabler. Poängen ökar det vill säga variabelns värde **ändras** alltid då spelaren lyckas ta fast skatten. **Sätt** i början av spelet poängen till värdet 0.



Om tiden är begränsad kan ni avsluta programmeringen av spelet här och resten av tiden kan deltagarna testa sina spel och till exempel ändra Spelarens rörelsehastighet (det vill säga ändra värdet med blocket "ändra x/y med") eller poängens tillväxthastighet (det vill säga värdet i blocket "ändra poäng med")

6. **Skapa en ny sprajt: Fiende och namnge den.**

7. Fienden pekar mot spelaren och rör sig sedan framåt. Detta upprepar den under hela spelet i blocket "för alltid".



8. När fienden får fast spelaren slutar spelet, det vill säga hela programkoden stoppas.



9. Testa spelet och låt fienden fånga spelaren. När du testar spelet på nytt händer sannolikt ingenting längre! Detta beror på att fienden börjar på spelaren, det vill säga aktiveringen av spelet i punkt 8 aktiveras genast.

Vi ska alltså ställa in utgångspunkten för spelaren och fienden. Dra spelaren med musen till något hörn till utgångsläget. Ange sedan som första kommando i skriptet som gjorts i punkt 8 blocket "gå till".

Gör det samma för fienden.

Obs! Talvärdena för blocket "gå till" behöver inte vara desamma som nedan, utan talvärdena för blocken i verktyglådan uppdateras av sig själva så att de passar för ditt spel alltid då du drar sprajten till en ny punkt. Nedan ett exempel på placeringen av blocken i kommandoserier.



Spelare



Fiende





TREDJE VERKSTADEN

Astronomerna

I rymden finns många olika himlakroppar, såsom planeter, stjärnor och asteroider. Asteroiderna är steniga himlakroppar som kan upptäckas från jorden som små ljuspunkter med starka teleskop. Man kan försöka fastställa asteroidernas form med hjälp av solljuset som reflekteras från dem. Kliv in i en astronoms stövlar och res på en spännande rymdfärd. Undersök vilka objekt de givna skuggbilderna kan föreställa och avbilda ett fiktivt solsystem.

Du ser endast skuggbilderna av de fiktiva asteroiderna. Din uppgift är att dra slutsatser om vilken himlakropp skuggbilden föreställer och hur man kan bygga den av ärter och stickor. I programmeringen fortsätter vi i formernas värld och kodar ett eget solsystem där planeterna rörs i banor som har formen av geometriska figurer (triangel, kvadrat, cirkel).



LITTERATURTIPS

- Camilla Linde: **Snack Parrow's intergalaktiska rymdbyrå** för underliga mysterier och piratbestyr
- Christer Fuglesang: **Rymdresan: Markus och Marianas äventyr med farbror Albert**
- Erik Mellgren: **Mars: din guide till den röda planeten**
- Dominic Walliman & Ben Newman: **Professor Astrokatt i rymden**

MYSTISKA OBJEKT

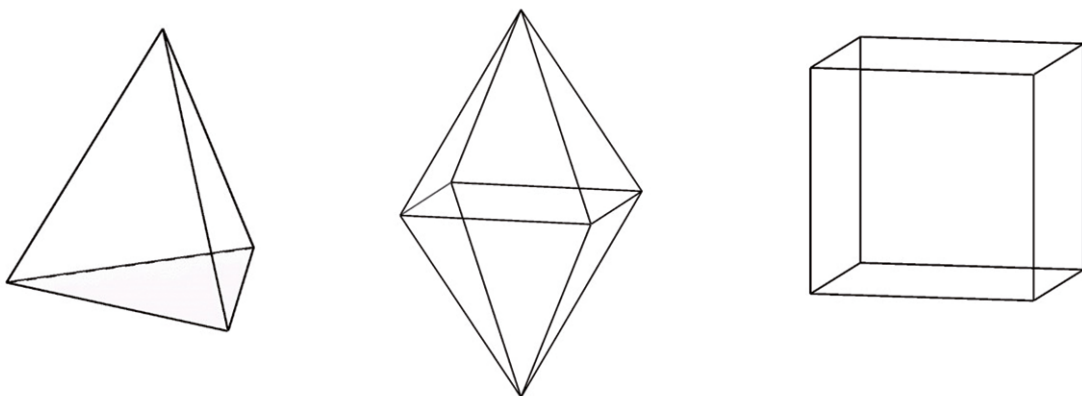
Tillbehör

- 1 påse (å 500 g) torkade örter
(blötlägg över natten före användning – du kan använda samma örter som för Volymutmaningen vid den första verkstaden)
- 1 000 st. cocktailstickor
- djupa papperstallrikar eller andra kärl
- kartong
- lim
- saxar
- skuggbildsvideo (finns på adressen: <https://pulmaario.luma.fi/>)
- pyssmall för tetraeder (som bilaga på sidan 83)
- pyssmall för hexaeder det vill säga kub (som bilaga på sidan 84)
- pyssmall för oktaeder (som bilaga på sidan 85)

Förberedelser

Blötlägg örterna över natten i rikligt med vatten. Fördela örterna och cocktailstickorna på djupa papperstallrikar så att varje grupp på 4–6 personer får en tallrik med örter och en med cocktailstickor.

Gör kartongmallar för de Platonska kropparna. Kropparna är sammanlagt fem men pyssla de tre lättaste av dem: tetraedern, oktaedern och hexaedern, det vill säga kuben. Kropparna består av enbart antingen regelbundna trianglar eller kvadrater.



De Platonska kropparna tetraedern, oktaedern och hexaedern, det vill säga kuben.

Inled pysslandet genom att kopiera följande bilagor: mallar för tetraeder, hexaeder och oktaeder. Limma mallarna på kartong eller annat hållbart papper. Klipp varje figur noggrant. Vik längs strecken på figurerna. På figurerna finns nummer där du ser vilka delar som ska limmas på varandra, det vill säga ettorna på varandra, tvåorna på varandra och så vidare. Den mörkare figuren kommer alltid längst ner. Stryk lim på det mörka området och limma ihop det med motsvarande nummer. Det lönar sig att sätta ihop kropparna i nummerordning. Håll kartongmallarna gömda i början av verkstaden.

**DET GÅR ATT GÖRA
PLATONSKA KROPPAR OCKSÅ
AV KASSERADE BÖCKER.
LIMMA PYSSELMALLEN
PÅ EN BOKSIDA OCH
PYSSLA KROPPEN ENLIGT
ANVISNINGARNA.**

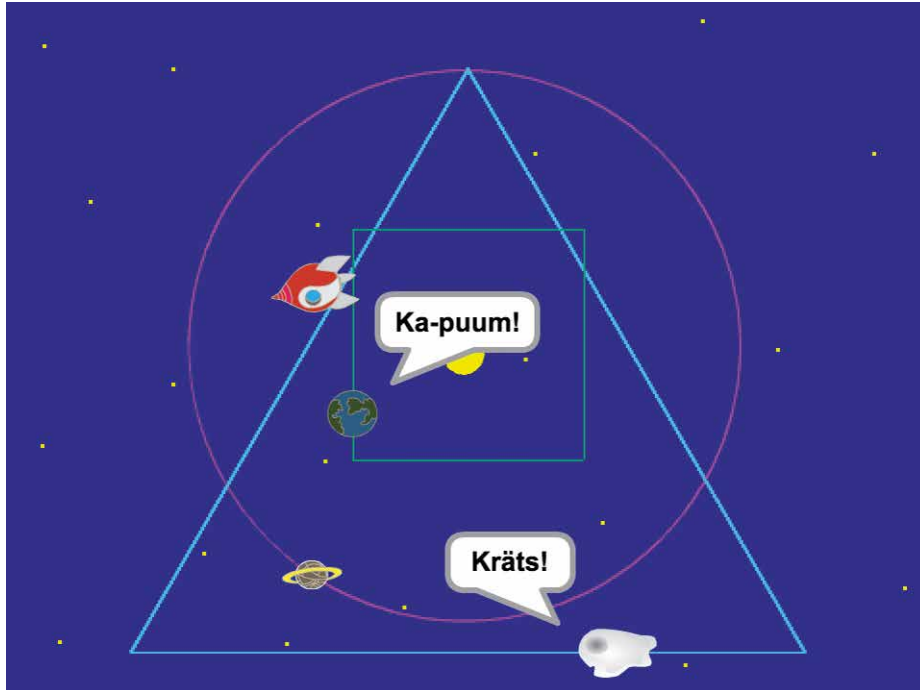


Uppgiftsanvisningar

Bygganvisningar

1. Öva första att bygga med stickor och ärter. Be deltagarna att först bygga en triangel, kvadrat och femhörning.
2. Visa sedan tetraeder-videon (video 1). Uppmuntra till att bygga en egen kropp istället för att ta modell av en kompis. Alla uppfattar kropparna på olika sätt och kompiserna kan se kroppen på ett annat sätt.
3. Då alla har byggt sin version av kroppen, avslöja rätt kropp genom att visa upp tetraeder-kartongmodellen.
4. Visa sedan hexaeder-videon (video 2) och avslöja igen rätt kropp då deltagarna är färdiga.
5. Visa ännu oktaeder-videon (video 3) och visa även kartongmodellen på en oktaeder när deltagarna har byggt färdigt.
6. Till slut kan du låta deltagarna fritt bygga de figurer och kroppar som de vill.

SOLSYSTEMSPELET



I solsystemet roterar planeterna runt solen och tillkännager kollisioner på sina egna sätt. Planeterna i detta solsystem roterar inte i vanliga ellipsbanor utan kvadrat-, triangel- och cirkelbanor.

Dessutom finns ett rymdskepp som styrs av spelaren.

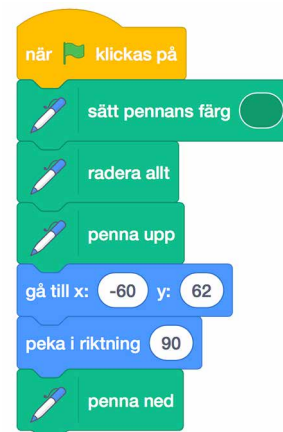
1. **Skapa en ny sprajt: Jorden**, och namnge den (se s. 17). Sprajten behöver inte se ut som ett jordklot utan deltagarna kan använda sin egen fantasi.

I det här projektet behöver du Penna-tillägget. Tilläggen finns i programmeringsvyns nedre vänstra hörn.



2. Jordklotet roterar i en kvadratisk bana. Flytta jordklotet först med musen till det fiktiva vänstra övre hörnet. Koordinaterna i verktygslådans "gå till x:_y_"-block ändras automatiskt alltid då sprajten dras till ett nytt ställe på spelområdet. Detta block dras alltså till programmeringsområdet först då sprajten redan är på plats på sin bana i det övre vänstra hörnet.

Jordklotet roterar i sin bana medsols, så placera det att peka mot höger (90°). För att banan ska bli snygg på spelområdet, radera spelområdet först så det blir tomt och lyft upp pennan från "pappret" innan du förflyttar dig till startpunkten. Sätt ner pennan då förberedelserna är gjorda.



3. Då du ritar en fyrkant ritar du sidan och hörnets vändning (rät vinkel, dvs. 90°) fyra gånger. Sidan kan ritas med ett "gå 120 steg"-block men då ser du inte när sidan ritas. Sidan ritas lite långsammare om du repeterar rörelsen i små steg, till exempel 10 gånger 12 steg.

För att jordklotet inte ska stanna genast efter det första varvet, sätts kommandoserien som ritar kvadraten in i för alltid-slingan. Lägg till ritandet av kvadraten efter förberedelserna för den föregående punkten.



4. **Skapa två nya sprajtar: Komet och Saturnus** och namnge dem. Lägg till kometen på triangelbanan och Saturnus på cirkelbanan. Placera kometen i övre hörnet av en fiktiv triangel och Saturnus i cirkelns vänstra kant.

Teori: Triangelns vinkel är 60° , men "pennan" måste vändas som dess **supplementvinkel** det vill säga $180^\circ - 60^\circ = 120^\circ$. Du får en cirkel då du gör en vändning på 1° 360 gånger, eftersom en full cirkel är 360° .

I ju fler repeteringar rörelsen har delats in desto långsammare är satelliten. Antalet steg inverkar på figurens storlek.



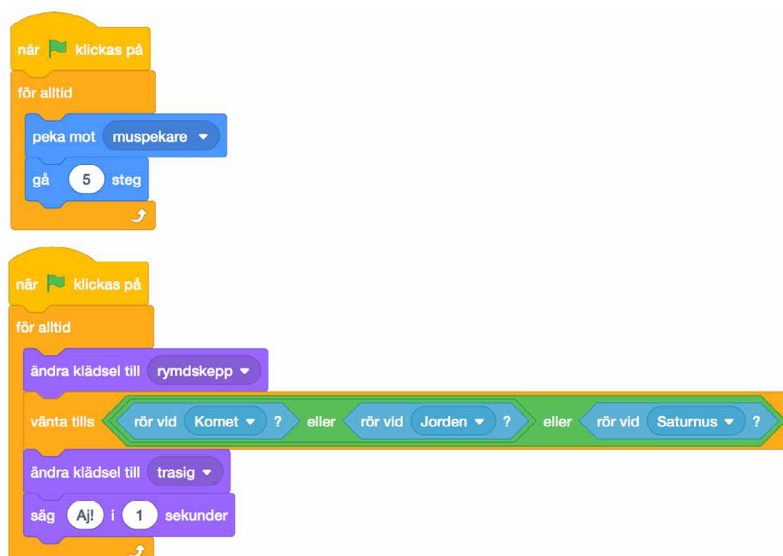
a) Komet



b) Saturnus

5. **Skapa en ny sprajt: Rymdskepp.** Rymdskeppet ska följa musen under hela spelet, så därför använder vi igen blocket för alltid. Programmera rymdskeppet att säga till exempel "Aj!" alltid då det krockar med någon av planeterna.

Valfritt: Om det ännu finns tid i verkstaden i det här skedet, **skapa två olika klädselar** för rymdskeppet, en av det hela rymdskeppet och en av ett som fått en träff. Ge spelet kommandot att byta till den hela klädseln i början av spelet. En klädsel som träffats byts ut vid en kollision. Man återgår till den hela klädseln då rymdskeppet har klarat kollisionen. Detta repeteras under hela spelet.

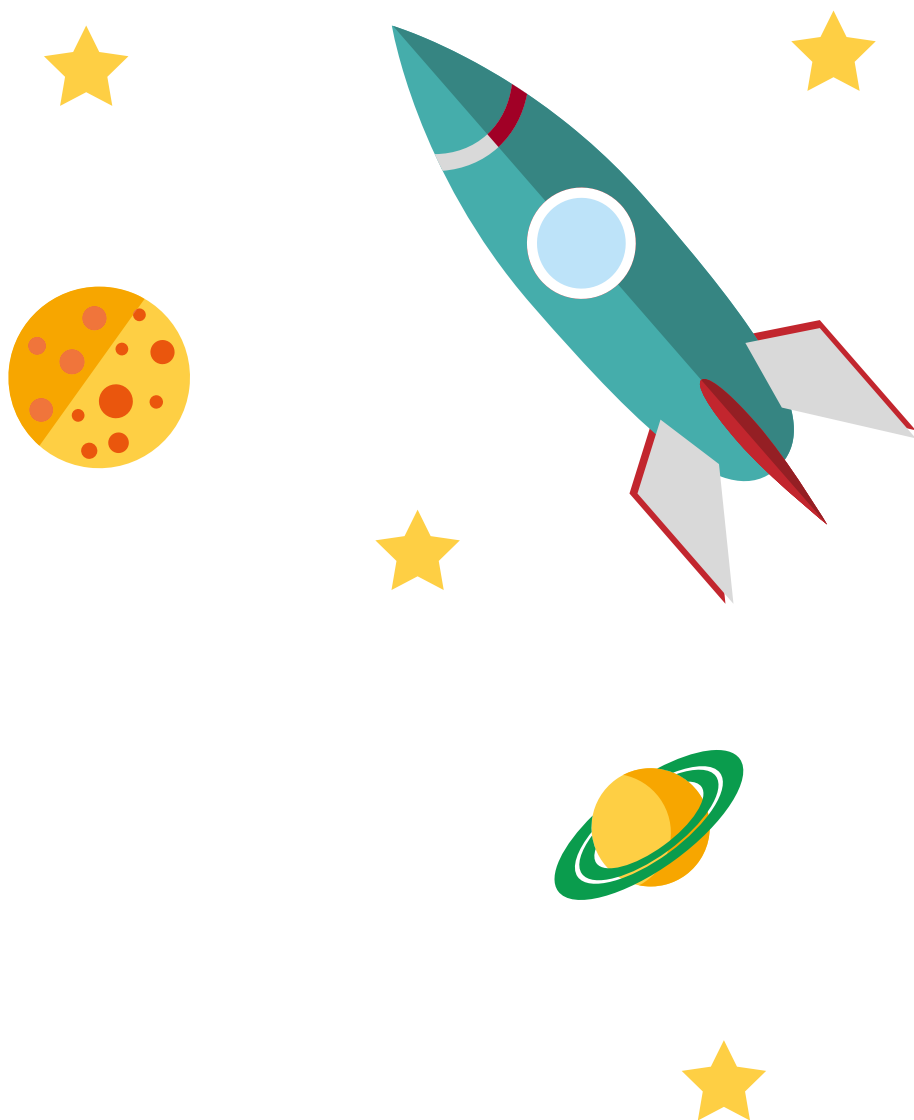


6. Valfritt: Lägg till lämpliga variationer av krockarna för planeterna. Till exempel Saturnus kan alltid vänta tills den krockar med någon annan sprajt och sedan säga till exempel "Krasch" en sekund.



7. Valfritt: **Skapa en ny sprajt: Sol.** Skapa två klädselar för Solen, så det ser ut som den skiner när de byts ut. Roterar också om du vill Solen hela tiden på plats.





L

R

A



T



N

C

X

Ö

R

S

V



FJÄRDE VERKSTADEN

Detektiverna

Vid detektivbyrån finns det mycket att göra och brott som måste lösas. Denna gång har detektiverna fått oklara meddelanden vars innehåll är rotvälska! Hemlighållandena måste lösas så man kommer tjuvarna på spåren. Vad tänker tjuvarna råna härnäst? Lyckas bakhållet och får ni fast tjuvarna på bar gärning?

Verkstadens tema är detektiver. I matematikuppgiften bekantar ni er med Caesar kryptering som ni reder ut med hjälp av krypteringsskivan. Ett krypterat meddelande styr er tjuvligan på spåren och gripandet sker genom programmering. Tjuvarna grips med det och detektiverna måste kunna förutse var nätet fallet.



LITTERATURTIPS

- Arthur Conan Doyle: **Sherlock Holmes** (tecknad serie)
- Enid Blyton: **Fem reser till havet**
- Beppe Singer: **Beppe testar! 15 detektivsexperiment**
- **Detektivens handbok**

KRYPTERADE MEDDELANDEN

Tillbehör

- kartong
- grennitar (1 st. per deltagare)
- pennor
- papper
- saxar
- lim
- (passare)
- mall för krypteringsskiva (som bilaga på sidan 86)
- krypterade meddelanden 1 (som bilaga på sidan 87)
- krypterade meddelanden 2 (som bilaga på sidan 88)
- svar på de krypterade meddelandena (som bilaga på sidan 89)

Förberedelser

Kopiera för varje deltagare bilagorna krypteringsskivan samt krypterade meddelanden 1. Kopiera dessutom bilagan krypterade meddelanden 2 för varje grupp och klipp loss uppgiftens ord från pappret.

Kopiera och pyssla ännu en krypteringsskiva som modell så att du kan visa den för deltagarna (se noggranna pysselanvisningar nedan).

De rätta lösningarna på de krypterade meddelandena finns som bilaga.

Uppgiftsanvisningar

Pysselanvisningar

1. Varje deltagare gör en egen krypteringsskiva. Pysslandet av skivorna inleds med att limma skivans delar på kartong och klippa loss delarna från kartongen.
2. Mitt på skivorna görs noggrant hål med antingen en passare eller penna. Säg till deltagarna att de gör hålen mycket noggrant så att skivorna passar exakt på varandra. Detta gör det betydligt enklare att avläsa skivan.
3. Skivorna läggs på varandra så att den mindre skivan är ovanpå. Skivorna sätts ihop med grennitar. Grenniten trycks igenom hålet och nitens vingar viks på baksidan av den större skivan.
4. Nu är skivan färdig!

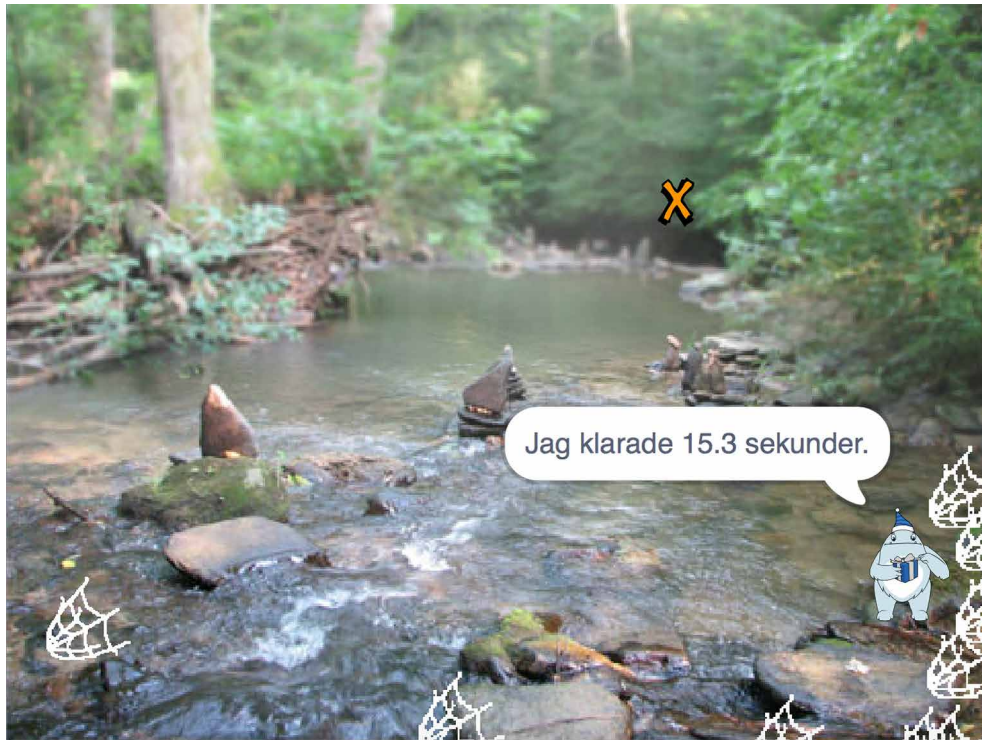
Att lösa meddelandet

5. Öva att använda krypteringsskivan genom att först lösa färdiga krypterade meddelanden med hjälp av uppgiften krypterade meddelanden 1. När orden har lösts, dela in deltagarna i grupper på 3–4 personer och ge varje grupp i uppgift att lösa orden i krypterade meddelanden 2. Det är meningen att med orden från krypterade meddelanden 2 bilda en vettig mening på svenska.
6. För att kunna lösa meddelandet behöver du känna till krypteringsnyckeln. Krypteringsnyckeln är ett tal mellan 1 och 29 och det finns efter meddelandet i parentes. Till exempel det krypterade meddelandet LYRH (5) betyder ”hund”.
7. För att lösa ett meddelande som någon annan krypterat ska du snurra den mindre skivans mörka A-bokstav till motsvarande siffra på den större skivans krypteringsnyckel.
8. Håll skivorna noggrant i samma ställning i förhållande till varandra. De får inte längre snurra i det här skedet. Sök den första bokstaven i det krypterade meddelandet på den större skivan. Det ursprungliga meddelandets bokstav finns på samma ställe på den mindre skivan.
9. Fortsätt lösa meddelandet en bokstav i taget tills du har löst hela meddelandet.
10. Då du löst alla krypterade ord ska du ännu bilda en mening av dem. Var tänker tjuvarna slå till?

Kryptering av ett meddelande

11. Prova ännu på att kryptera egna meddelanden. Varje deltagare kan kryptera ett ord eller ett kort meddelande och sedan ge det till kompisen som får lösa det.
12. Krypteringen av ett meddelande sker i motsatt följd som lösande. Välj först krypteringsnyckel och snurra den mindre skivans A-bokstav till den siffra som motsvarar den större skivans krypteringsnyckel.
13. Sök den första bokstaven för meddelandet som ska krypteras på den mindre skivan. Bokstaven för det krypterade meddelandet är bokstaven på den större skivan i motsvarande punkt.
14. Fortsätt så här tills hela meddelandet är krypterat.

NÄTKASTSPELET

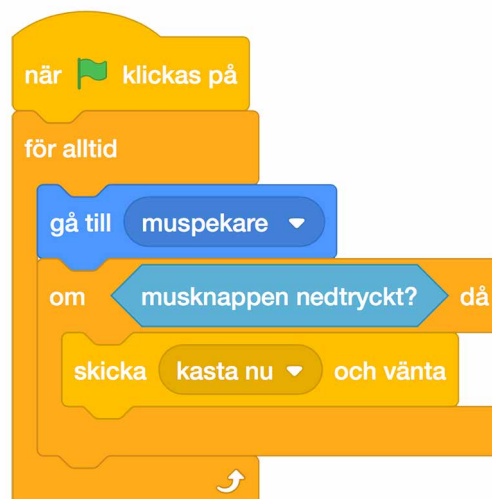


Spelaren siktar med krysset och försöker träffa tjuven. Nätet kastas genom att klicka med musen. Nätet flyger i en båge enligt den givna hastigheten och faller med inverkan av tyngdkraften.

När tjuven blir fast meddelar den hur lång tid flykten tog och rymmer på nytt.

1. **Skapa en ny sprajt: Sikte.** Som sikte passar till exempel bokstaven X i Scratch sprajtbibliotek. Förminska siktet så det blir ungefär två gånger så stort som muspekaren. Namnge sprajten (se s. 17).

2. Programmera siktet att **för alltid gå till musens markör**. Om du klickar på musen, skickas meddelandet "Kasta nu" (koden ska också vänta). Du kan ge meddelandet ett nytt namn genom att klicka på kommandots lilla triangel och välja "Nytt meddelande".



3. **Skapa en ny sprajt: Nät** och namnge den. Som nät passar av de färdiga sprajtarna till exempel spindeln, bläckfisken eller krabban. Ställ in utgångspunkten exakt $x = -200$ och $y = -120$ så att senare uträkningar fungerar.



4. Klicka på den gröna flaggan så att spelet startar och nätet flyttas till rätt punkt. Stoppa spelet och förminska nätet tills det med säkerhet **inte rör vid kanten**. Nätet programmeras senare att stanna när det vidrör kanten, varför det är viktigt att ändra på nätets storlek.

5. Då nätet upptäcker meddelandet "Kasta nu" ska det först räkna nätets hastighet. Hastigheten beror på hur långt borta siktet är då kastet går. **Skapa nya variabler: horisontalfart och vertikalfart** för alla sprajtar. Variabeln "slumpmässig styrka" är valfri i formlerna för horisontal- och vertikalfarten. I stället för den kan du ange siffran 13 (så att kastet skalas lämpligt på spelområdet). Ordningen på de gröna blocken inverkar på räkneordningen, försäkra dig om att subtraktionen sker i divisionen.

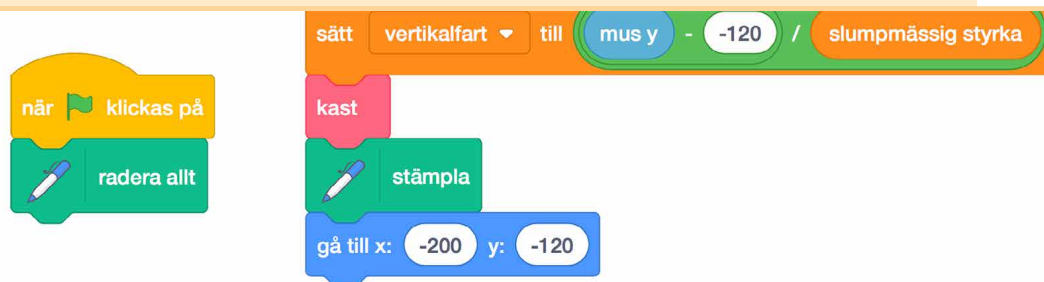


6. **Skapa ett nytt block: Kast.** Skapa ett block för nätsprajten. Den rosa Mina block-sektionen finns i botten av verktygslådan. Det nya blocket behöver bara ett namn, ingen indata eller något annat. Kastet genomförs på samma sätt som en animation i små snuttar där x- och y-värdena ändras lite åt gången. Efter varje liten snutt påverkar tyngdkraften vertikalfarten. Kastet tar slut då nätet träffar kanten.

Sätt ännu till ett nytt "Kast"-kommando från menyn Mina block efter bestämmelserna för hastigheten.



7. Ställ ännu till slut in nätet att **stämpla sig själv** så att spelaren minns slutresultaten av sina kast. Här behöver vi Pennan igen. Se [sidan 51](#). Efter det flyttar sig nätet igen till utgångspunkten. I början av spelet raderas spelområdet så att kasten från det förra spelet inte längre syns.

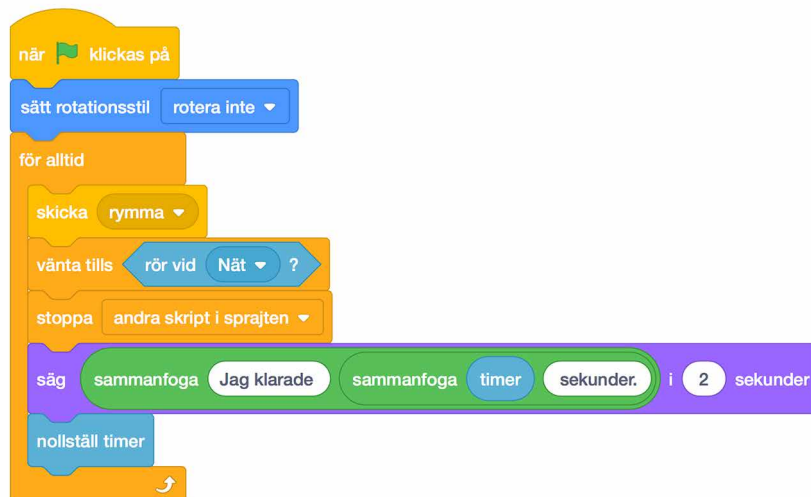


8. **Skapa en ny sprajt: Tjuv.** Dra tjuven till spelets högra kant. Säkerställ att tjuven inte **träffar i någon kant**. I det här spelet kan tjuven inte heller vända sig så ställ in rotationsstilen till att börja med. Skicka efter det här meddelandet "Rymma". Då tjuven tar emot meddelandet börjar den hoppa undan detektiverna i den övre kanten. Det är viktigt att tjuven först pekar rakt uppåt annars kan den röra sig mot den vänstra kanten och det skulle vara för lätt att få fast den.



9. Fortsätt skriptet i föregående punkt som syns i bilden till vänster. Först väntar tjuven på att fångas av nätet (skriptet för att rymma rullar samtidigt hela tiden på i bakgrunden). Då tjuven fastnar i nätet ska du stoppa sprajtens övriga skript, det vill säga i praktiken skriptet rymma. Ställ sedan in tjuven att säga hur lång rymmarstråten var.

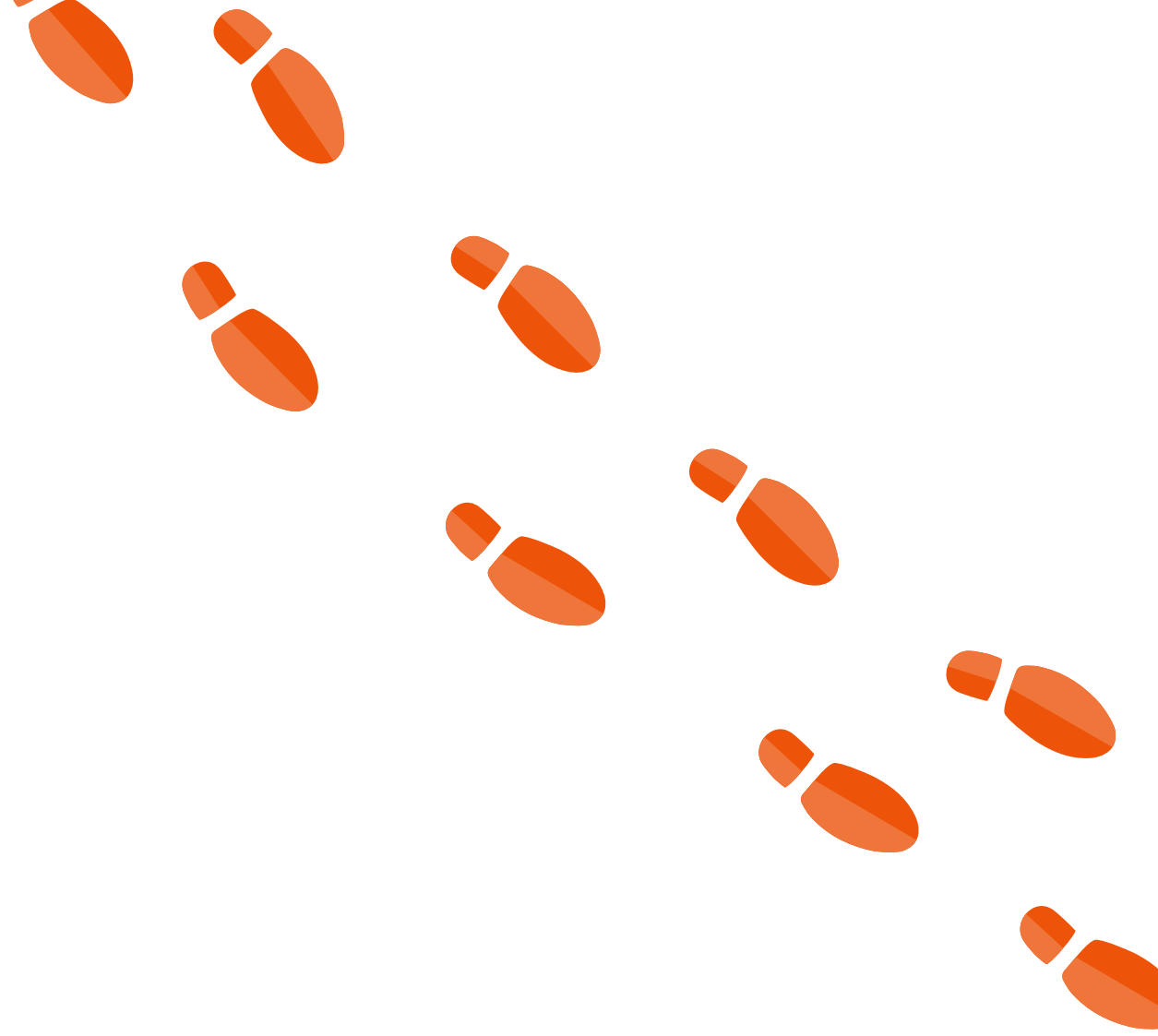
Om du vill att det ska gå att ta fast tjuven flera gånger under samma spel ska du ännu nollställa timern och sätta blocket "för alltid" runt hela funktionsserien (se bild). Nu börjar flykten igen från början då tjuven har tagits fast.

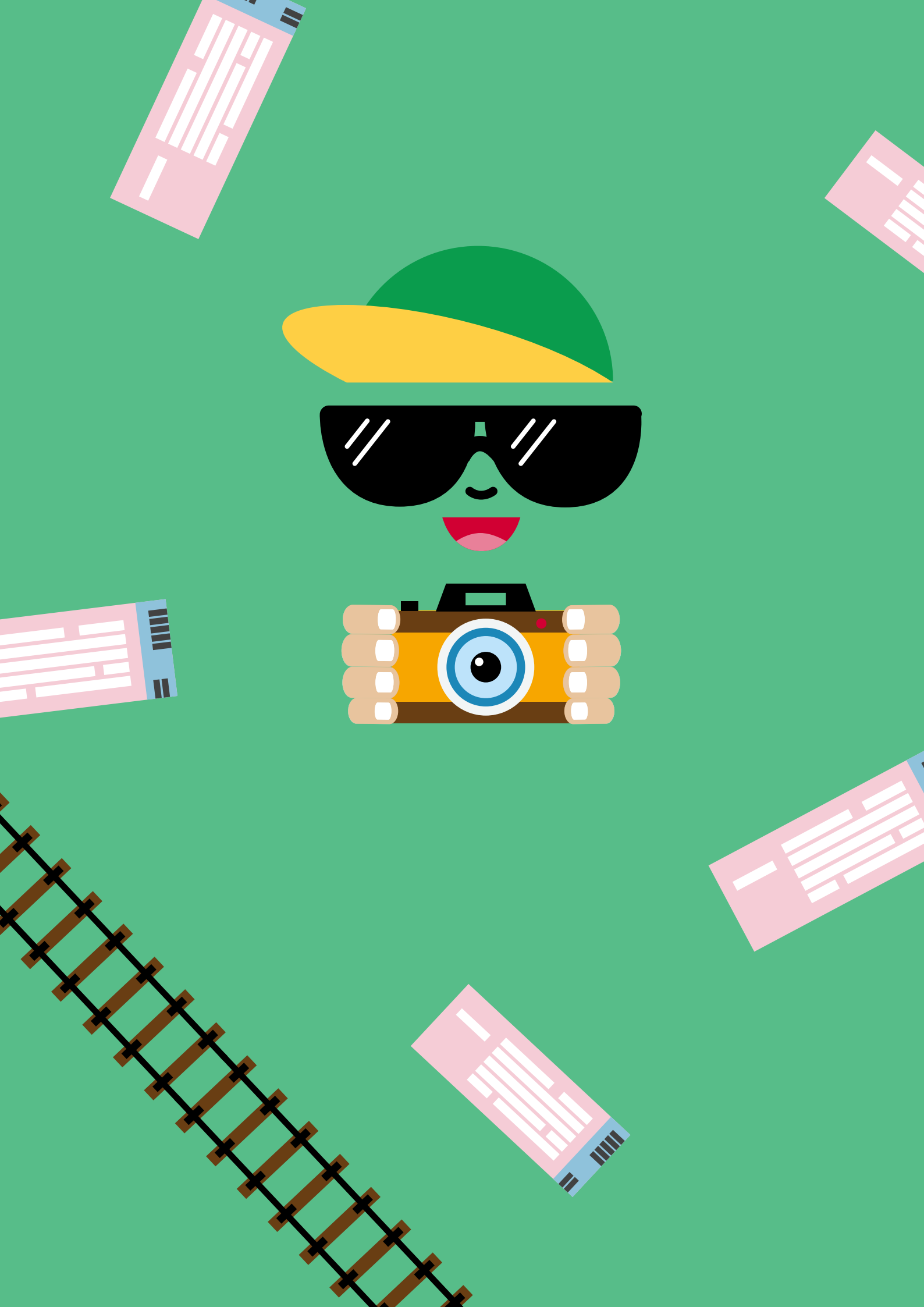


Ytterligare information:

Blocket är ett av programmeraren själv definierat nytt kommando. Då kommandot "block" genomförs i programmet fortsätter programmet med blocket till följande kommandon först när kommandona som definierar blocket har utförts. I detta avseende motsvarar det kommandot "skicka meddelande och vänta".

Blocket hör alltid till en viss sprajt. Då block-kommandot används reagerar de andra sprajterna inte på det även om man skulle ha definierat ett block med samma namn för dem. Blocket kan alltså inte användas från en annan sprajt.





FEMTE VERKSTADEN

Jorden runt- resan

Ni är på väg på en utlandsresa. På tågbiljetten som du håller i står det till vilken stad du ska resa. Ni kommer att åka tåg. Järnvägsnätet förenar länder från Europa till Asien men nätet är redan gammalt och spåren trånga. Bara ett tåg åt gången kan köra på spåret. Förutom på spåren är det även trångt på tågstationerna och det finns endast rum för en person åt gången på dem. Hur ska de ivriga resenärerna komma tryggt fram? På returren tar alla samma tåg så att de kan berätta för varandra om sina resor.

Det matematiska problemet i verkstaden är ett spel med järnvägsnät som utvecklar problemlösningsfärdigheterna och där samarbete är en nödvändighet för att lösa problemet. Kommunikationsmedlet för returren görs genom att programmera i stil med det traditionella maskspelet.

LITTERATURTIPS

- Charlotta Lannebo: **Lilla E i Karibien**
- Sören Olsson & Anders Jacobsson: **Sune på bilsemester**
- Erik Mellgren: **Mars: din guide till den röda planeten**
- Moa och Molle reser jorden runt

JÄRNVÄGSNÄTET

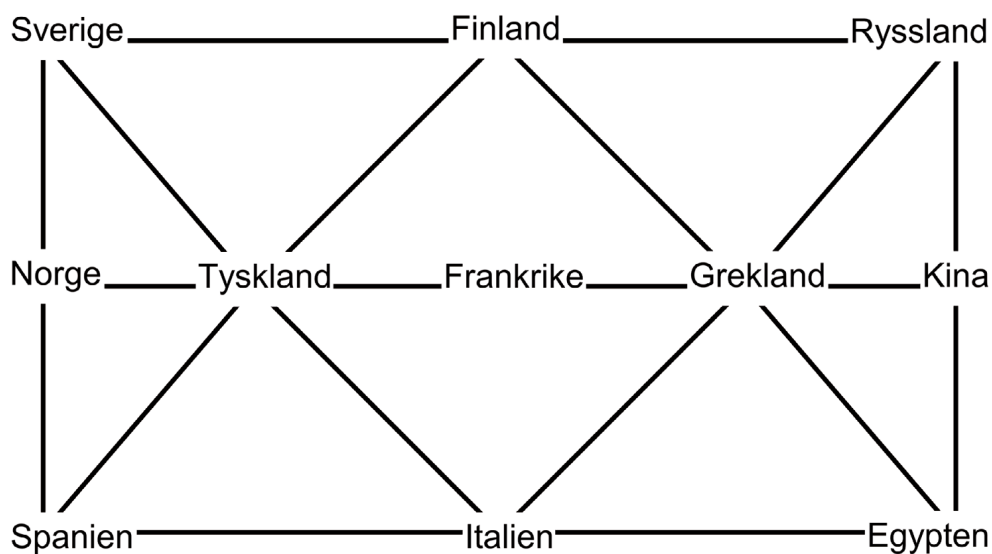
Tillbehör

- 1 rulle målartejp (bredd ca 1–2 cm)
- stor spelbricka (t.ex. bok eller kramdjur)
- pappersark A4
- penna

Förberedelser

Tillverka spelkort för Järnvägsnätspelet genom att klippa itu A4-pappren. För ett spel behövs sammanlagt 22 spelkort. Skriv på varje spelkort namnet på ett land eller huvudstadens namn. Länderna och huvudstäderna är: Finland, Helsingfors; Sverige, Stockholm; Ryssland, Moskva; Kina, Peking; Italien, Rom; Frankrike, Paris; Spanien, Madrid; Egypten, Kairo; Tyskland, Berlin; Norge, Oslo; Grekland, Aten.

Gör rum för spelområdet på golvet. För ett järnvägsnät behövs ett område på 2 m x 3 m. Tejpa ett nätverk enligt bilden med målartejp på golvet. Fäst **landskorten** i nätverket enligt bilden med målartejp.



Obs! I ett järnvägsnät finns det plats för högst tio deltagare åt gången. Om antalet deltagare är fler än tio lönar det sig att göra två separata nätverk.

Uppgiftsanvisningar

Spelets gång

1. Instruera deltagarna att ställa sig på landskorten i nätverket så att alla står på ett eget kort. Minst ett land ska förbli tomt.
2. Dela ut ett huvudstadskort till alla deltagarna. Huvudstaden på kortet får inte vara det landets huvudstad där spelaren står när spelet börjar. De överflödiga korten lämnas bort.
3. På kortet som spelaren håller i står det vart hen ska. Avsikten är att varje spelare ska kunna resa tryggt till sitt land. Det finns emellertid strikta regler för hur man kan röra sig i järnvägsnätet:
 - Det får finnas endast en spelare åt gången i varje land.
 - Bara en spelare får röra sig åt gången. Spelaren som rör på sig får spelbrickan som hen alltid ger vidare till följande spelare som är i turen.
 - Turen går till den som har möjlighet och för vem det lönar sig att röra på sig. Det är alltså endast en spelare som kan röra på sig längs spåret direkt till ett ledigt land.
 - Även om en spelare skulle nå sitt land kan det hända att hen ännu måste gå bort därifrån så att alla spelare når sina destinationer. Detta kräver samarbete!
4. Spelet är slut då alla spelare har nått sina destinationer.

DET GÅR OCKSÅ ATT SPELA JÄRNVÄGSNÄTET MED OLIKA TEMAN. TILL EXEMPEL KAN JÄRNVÄGSNÄTET BLI ETT SKORSTENSPULVERNÄT DÄR SPELARNÄR PERSONER I HARRY POTTER-BOKSERIEN SOM FÖRSÖKER NÅ SINA EGNA FAVORITPLATSER. TILL EXEMPEL CORNELIUS TOFFEES MÅL SKULLE VARA TROLLDOMSMINISTERIET OCH MOLLY WEASLY SKULLE VILJA TILL KRÅKBOET.



MASKSPELET



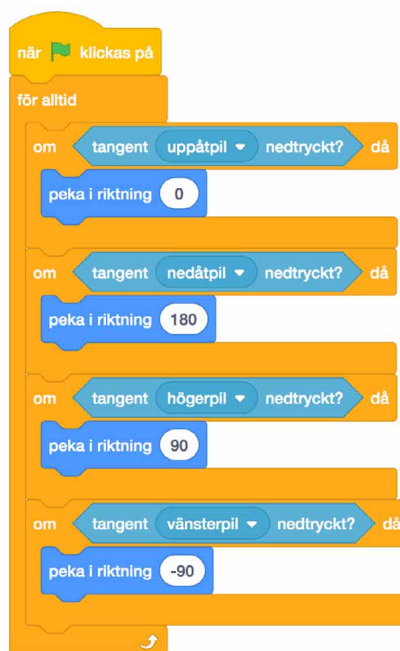
En helikopterkö samlar in resenärer och blir längre då den tar upp resenärer. Kön styrs med piltangenterna. Kön får inte krocka med sig själv.

Tanken i spelet är utmanande för det finns bara en helikopter och den lämnar kopior efter sig. Kopiorna rör sig inte utan iakttar när de inte längre hör till kön och försvinner.

1. **Skapa en ny sprajt: Kö.** Kö består i praktiken av små bitar så välj eller rita en sådan sprajt som är endast en del av kön, inte hela kön. Med tanke på köns utseende är det ofta bra om sprajten är ungefär lika bred som hög. Om du ritar köbiten själv är **det mycket viktigt att ställa in bildens mittpunkt rätt**. Bilden ska peka mot höger (i klädslar). Som bild passar till exempel alla de kommunikationsmedel som fyller kraven.

Namnge sprajten (se. s. [17](#)).

2. Programmera köändan att vända sig i samma riktning som pilen som trycks ned.



3. Programmera köändan det vill säga den ursprungliga sprajten att röra sig för alltid. Det lämpliga antalet steg kan variera, men till exempel **35** är en bra utgångspunkt. Sedan väntar köändan och skapar en kopia av sig själv.

Testa spelet. Köns som bildas borde vara enhetlig men inte för tät. Du kan ställa in antalet steg så det passar dig.





4. **Skapa en ny sprajt:**
Resenär Skapa även en ny variabel: **längd** som syns för alla sprajtarna.

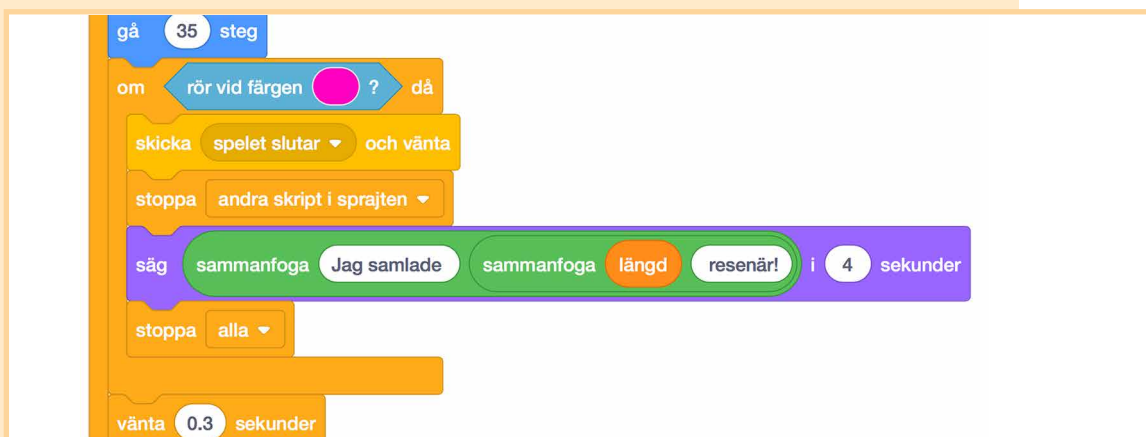
Programmera resenären att lotta en slumpmässig punkt för sig själv och vänta där på kön. Då resenären har stigit ombord ökar längden på kön. Längden är då spelet börjar 1 (dvs. endast köändan syns).



5. **Skapa en ny variabel för kön: "vilken är jag"** som är **enbart för denna sprajt**. När du skapar variabeln, välj "Enbart för denna sprajt". Då kan varje klon räkna ut sitt ordningsnummer. Ordningsnumret är i början 2 (ändan är det första) och ökar så ofta som kön rör sig. Om ordningsnumret blir större än köns längd måste klonen tas bort.

6. Gå till kösprajtens flik Klädslar. Sätt till en fläck med någon färg i **mitten** av köbiten så att det märks om kön krockar med sig själv. Denna färg **får inte finnas någon annanstans i spelet**, inte ens i bakgrunden. Sätt alltid i köns kommandon till ett test om den här fläckens färg vidrörs efter att kön rört sig. Instruktioner för hur det görs finns under kodexemplet.

Om det skedde en krock meddelas spelaren hur lång kön blev och spelet stoppas. På exempelkoden skickas också ett meddelande. Mer om detta i nästa punkt.



6.1 Färgkontroll

Genom att klicka på färgfläcken på kommandot "rör vid färgen" får du välja den färg som ska testas. På den ruta som öppnar sig finns det en pipettikon. Med en pipett är det möjligt att välja en färg från spelområdet. När man väljer färgen måste man vara ordentlig - ibland kan det även vara lättare att bara kolla på ritområdet vilken färg, mättnad och ljusstyrka som valdes för fläcken och välja desamma för kommandot.

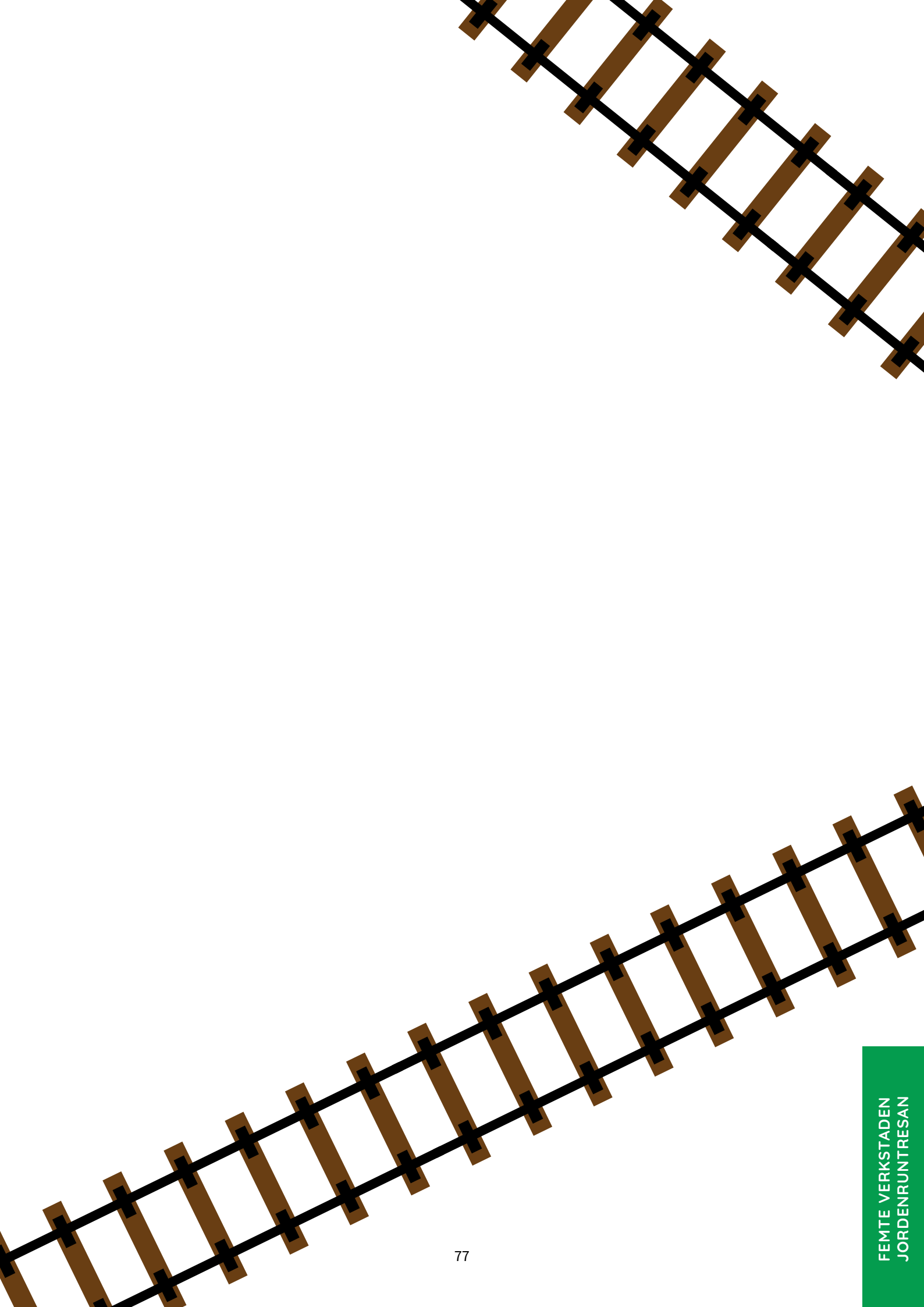


7. När spelet är slut ska du stämpla köns klonade biter på spelområdet, för klonerna försvinner då spelet stoppas. Här behöver vi Pennan igen. Se [sidan 51](#). Skicka ett meddelande för detta och vänta att det utförs (bild i föregående punkt). Kön tar också emot meddelandet (bilden nedan). Radera i spelets början stämplingsspåren. Du ska göra det i början av kommandoserien som innehåller testet om färgfläcken vidrördes.



8. Valfritt: Programmera kön att fortsätta från den andra sidan av spelområdet då köändan kommer till spelområdets kant.





Fler problem

Vill ni ordna fler Kluraklubb-verkstäder? Eller vill ni till exempel ha instruktioner till verkstäder för vilka det inte behövs datorer? På webbplatsen <https://pulmaario.luma.fi/> finns olika tilläggsmaterial som stöd för ordnandet av verkstäder samt bland annat en pdf-version av denna guide om ni behöver flera kopior.

LINKKI

Linkki är en vetenskapsklass vid Helsingfors universitets institution för datateknik som ordnar hobbyprogrammeringsverksamhet för elever i grundskolan och gymnasier. De klubbar, läger och evenemang som Linkki ordnar erbjuder ungdomar som är intresserade av datateknik en möjlighet till likasinnat sällskap och att få lära sig om programmering. Linkki är dessutom en kontaktpunkt för lärares kompletteringsutbildningsbehov och vi skapar utifrån vår hobbyverksamhet material som även andra kan använda. Linkki är en vetenskapsklass som lyder under Helsingfors universitets vetenskapsutbildningscenter och det nationella LUMA-centret.

Linkki erbjuder material för att inleda och komplettera en programmeringshobby. Förutom material om Scratch finns det material om många andra programmeringsmiljöer och lämpliga uppgifter för personer i mycket olika åldrar. Linkkis webbplats finns på adressen <http://linkki.cs.helsinki.fi/>.

SUMMAMUTIKKA

Summamutikka är ett center för matematikinläring som fungerar som en del av Helsingfors universitets vetenskapsutbildningscenter och LUMA-centret Finland. Summamutikkas mål är att inspirera och motivera barn och unga att studera matematik, stödja barns och ungas matematikstudier, matematikinläring och matematikhobbyer samt stödja matematiklärarna i deras viktiga undervisningsarbete och professionella utveckling. Summamutikka ordnar varierande klubb- och lägerverksamhet vars syfte är att erbjuda upplevelser av att lyckas för alla som lär sig matematik.

På Summamutikkas webbplats finns fler problem att lösa och spel att spela. Mer information om Summamutikkas verksamhet samt en materialbank för funktionell matematik finns på adressen <https://blogs.helsinki.fi/summamutikka/>.

Bilagor

Första verkstaden

- [Pappret för Volymutmaningen](#)

Andra verkstaden

- [Spelplan för Sänka skepp](#)

Tredje verkstaden

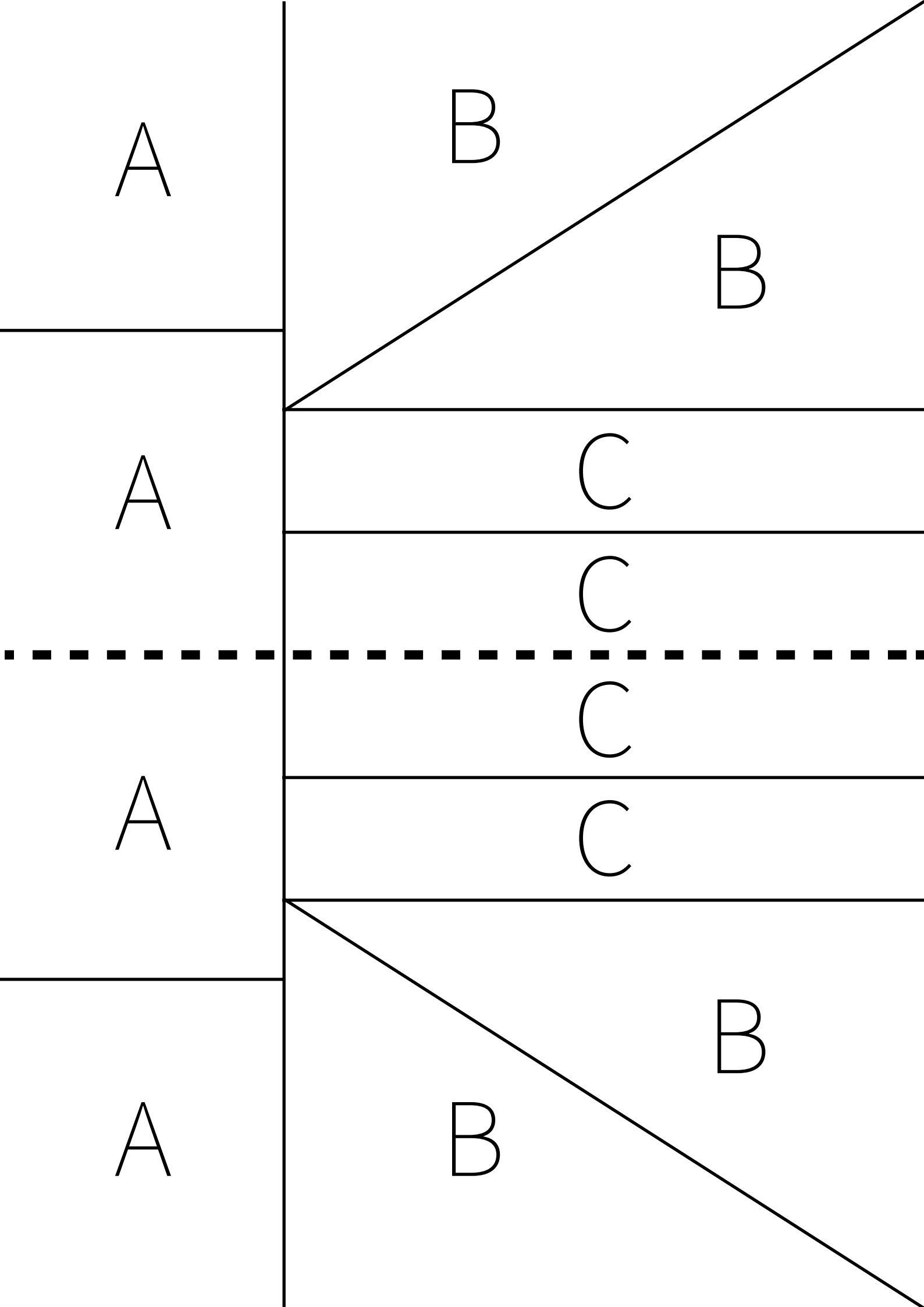
- [Pyssellmall för tetraeder](#)
- [Pyssellmall för hexaeder dvs. kub](#)
- [Pyssellmall för oktaeder](#)

Fjärde verkstaden

- [Underlag för krypteringsskiva](#)
- [Krypterade meddelanden 1 och 2](#)
- [Svar på de krypterade meddelandena](#)

Övriga bilagor

- [Programmering utan dator: CS Unplugged](#)
- [Underlag för övning med binära tal](#)
- [Diplom](#)



Spelplan för Sänka skepp

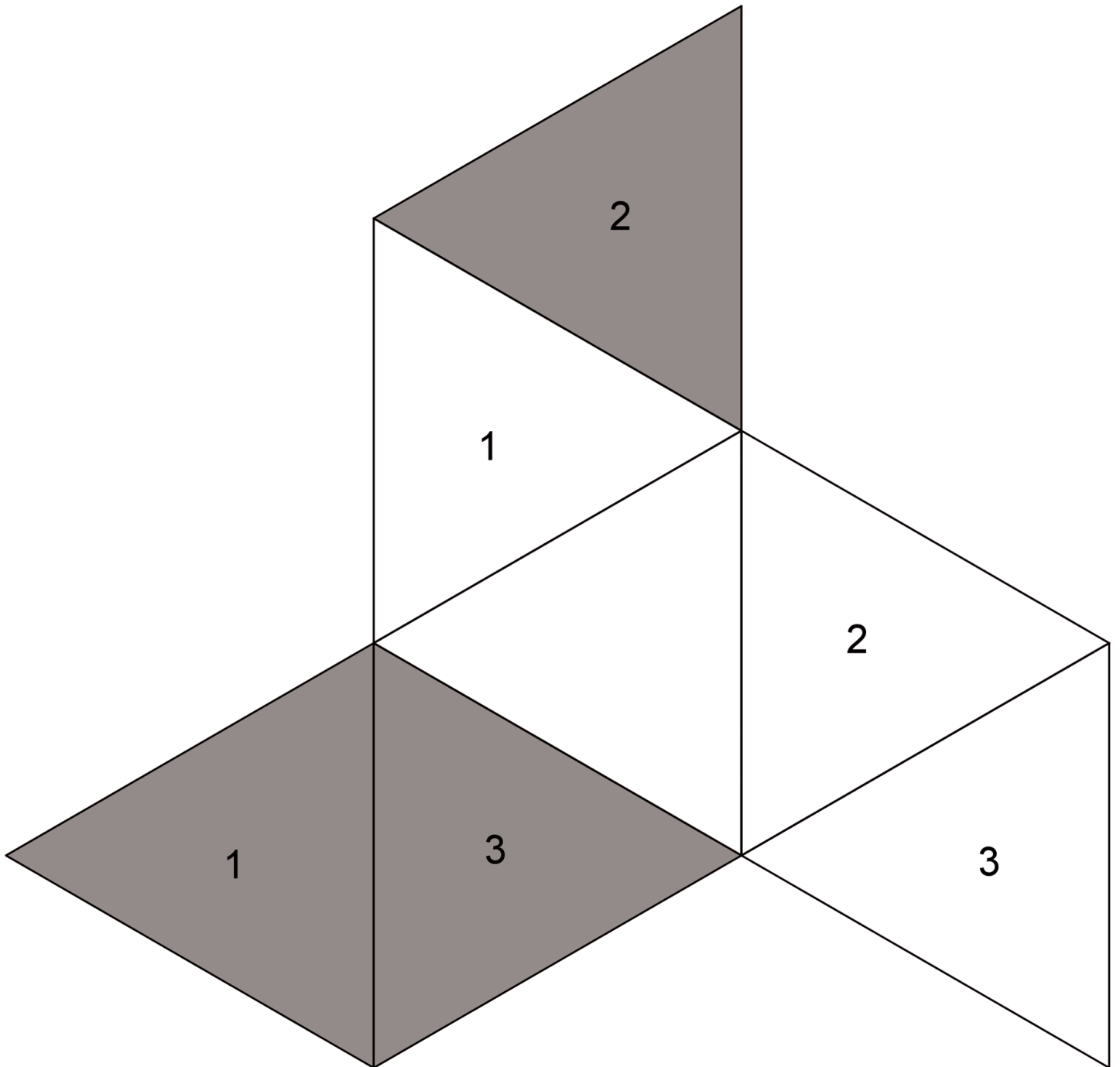
Ditt koordinatsystem

y9										
y8										
y7										
y6										
y5										
y4										
y3										
y2										
y1										
y0										
	x0	x1	x2	x3	x4	x5	x6	x7	x8	x9

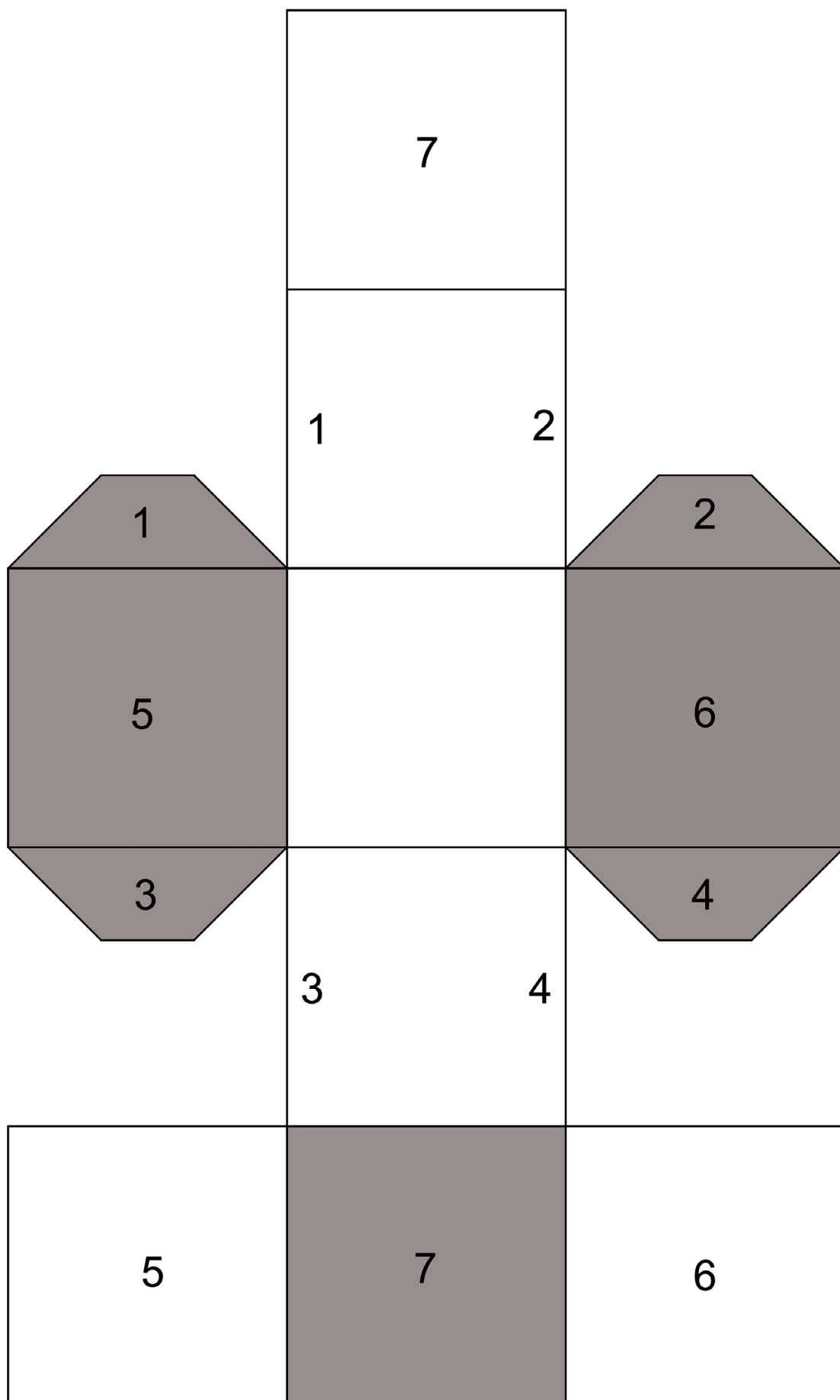
Kompisens koordinatsystem

y9										
y8										
y7										
y6										
y5										
y4										
y3										
y2										
y1										
y0										
	x0	x1	x2	x3	x4	x5	x6	x7	x8	x9

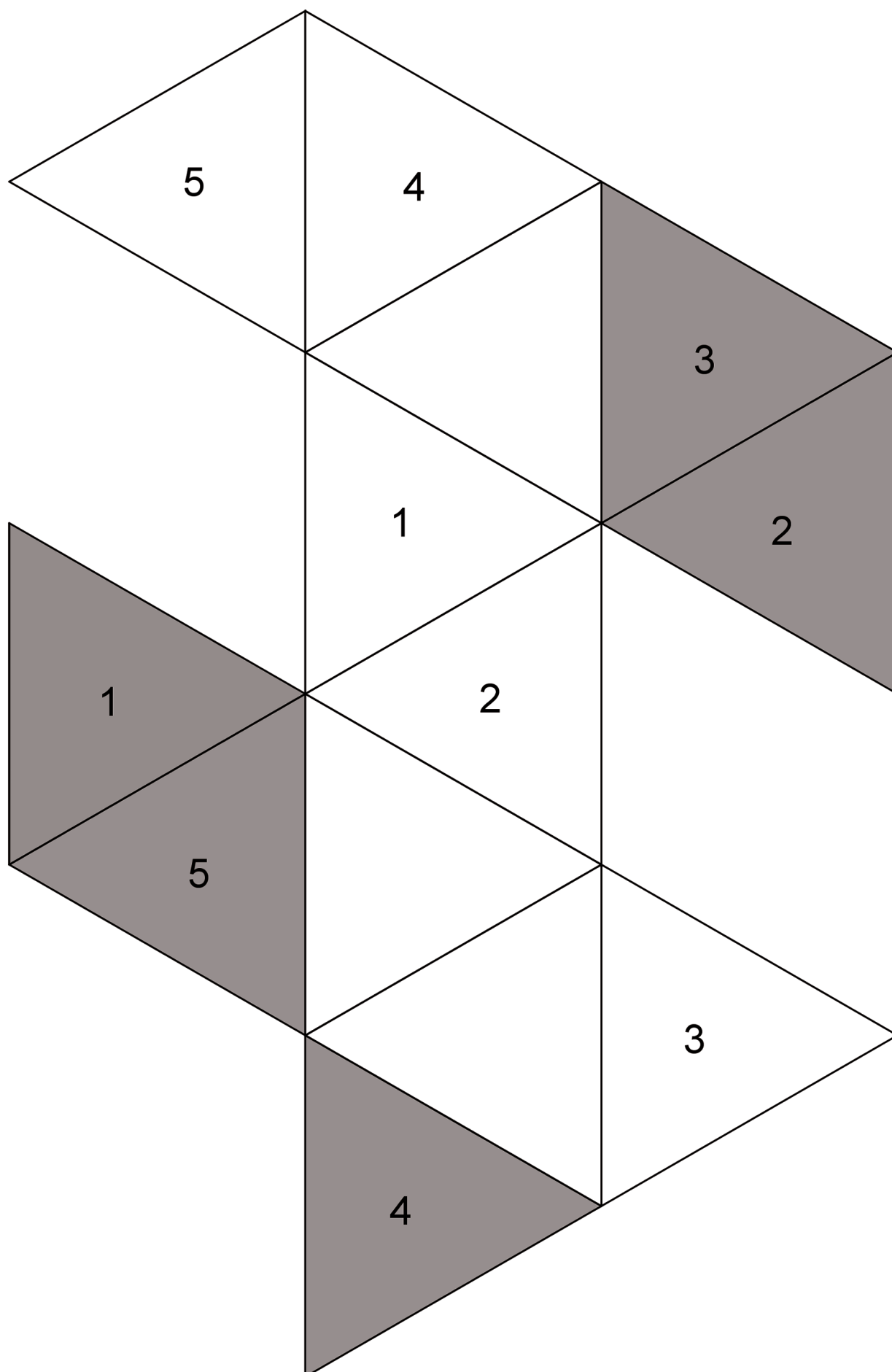
Pysselmall för tetraeder



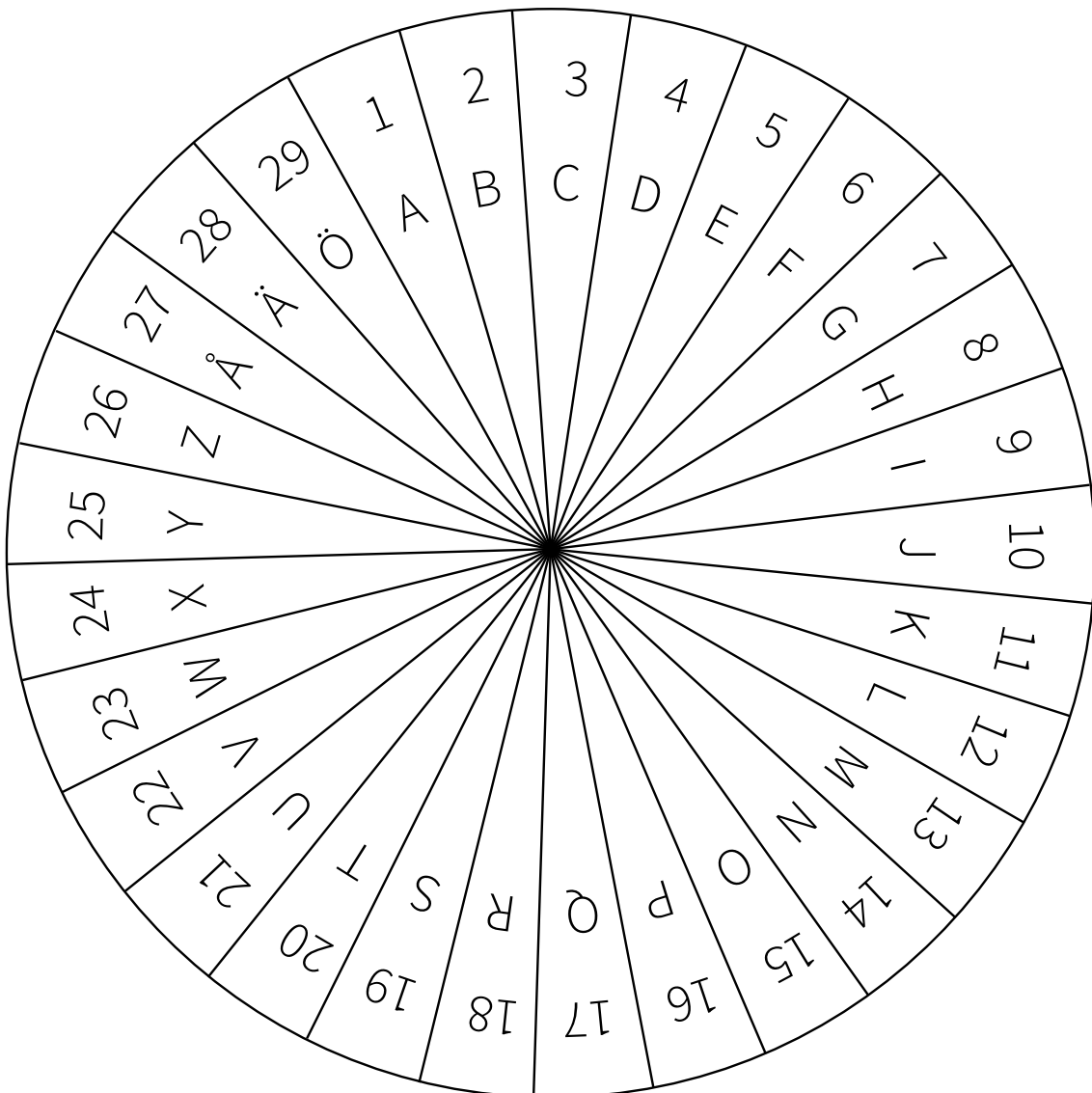
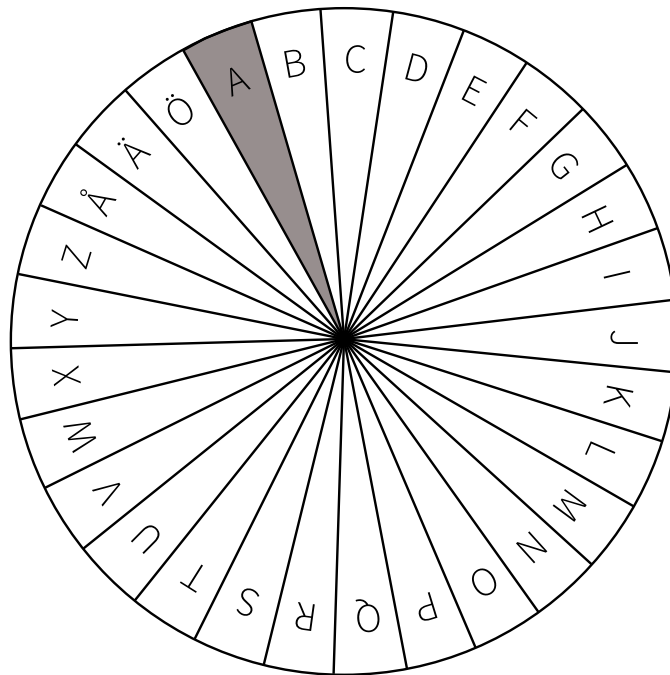
Pysselmall för hexaeder dvs. kub



Pysselmall för oktaeder



Mall för krypteringsskiva



En tre personers skurkliga har pinat detektiverna redan i flera månader. Vilka brott har tjuvarna hunnit begå hittills? Lös de krypterade orden och hjälp detektiverna att komma tjuvarna på spåren.



1. Vilket var skurkligans första rånmobjekt?

Svar: SQWÅS (9)

2. Vilket var tjuvarnas byte?

Svar: NSZVWLO (12)

3. Vilket var tjuvarnas senaste rånmobjekt?

Svar: ÖHFUHÖ (17)

4. Vad glömde tjuvarna kvar på brottsplatsen under det senaste rånet?

Svar: LERHWOEV (5)

5. Hur många tjuvar lyckades polisen få fast under det senaste rånet?

Svar: ACH (11)

Vad tänker tjuvarna råna härnäst? Lös orden i grupp och bilda av orden en mening på svenska så att ni kan ta tjuvarna på bar gärning.



UDTSPVEZHNAÅ (16)

YZKLAFVLAGF (22)

MX (27)

VWUDQGYLOOD (4)

TEYZG (7)

ÖS (2)

DEL 1

En tre personers skurkliga har pinat detektiverna redan i flera månader. Vilka brott har tjuvarna hunnit begå hittills? Lös de krypterade orden och hjälp detektiverna att komma tjuvarna på spåren.

1. Vilket var skurkligans första rånobjekt?
Svar: **KIOSK**
2. Vilket var tjuvarnas byte?
Svar: **CHOKLAD**
3. Vilket var tjuvarnas senaste rånobjekt?
Svar: **MUSEUM**
4. Vad glömde tjuvarna kvar på brottsplatsen under det senaste rånet?
Svar: **HANDSKAR**
5. Hur många tjuvar lyckades polisen få fast under det senaste rånet?
Svar: **TVÅ**

DEL 2

Vad tänker tjuvarna råna härnäst? Lös orden i grupp och bilda av orden en mening på svenska så att ni kan ta tjuvarna på bar gärning.

UDTSPVEZHNÅÅ (16)	lösning: FREDAGSKVÄLL
YZKLAFVLAGF (22)	lösning: DESTINATION
MX (27)	lösning: PÅ
VWUDQGYLOOD (4)	lösning: STRANDVILLA
TEYZG (7)	lösning: NÄSTA
ÖS (2)	lösning: ÄR

Orden bildar till exempel följande mening:

“Nästa destination är en strandvilla på fredagskväll.”

CS Unplugged är en webbplats där det finns mycket övningar om programmering och datalogiskt tänkande (på engelska). Du hittar den på adressen <http://csunplugged.org/>.

För dessa övningar behövs inte datorer. Många av dem liknar uppgifterna i den här guidens matematikavsnitt, men presenteras ur ett lite annorlunda perspektiv. Följande uppgift har fått inspiration av uppgiften Binary Numbers på webbplatsen CS Unplugged.

BINÄRA TAL

Tillbehör:

- **för varje deltagare en kopia på mallan till övningen med binära tal** (finns efter dessa anvisningar)
- **sax**
- **ett stort tomt papper samt en tuschpenna för handledaren**

Idén med uppgiften:

I uppgiften provar vi på att presentera tal i det binära talsystemet, det vill säga 2-systemet. Då siffrorna 0–9 används i tiosystemet används i det binära talsystemet endast 0 och 1. Detta är också idén med bitar som finns i datorerna – biten är antingen avslagen det vill säga 0 eller påslagen det vill säga 1. Om man till exempel vill skriva talet 2 i tiosystemet räcker inte en bit utan talet anges som 10 i det binära talsystemet.

Deltagarnas papperslappar representerar bitar. De kan vara antingen på det vill säga sidan med en punkt uppåt (1) eller av, det vill säga den vita sidan uppåt (0). När man räknar de enskilda synliga punkterna får man talvärdet som bitarna representerar i tiosystemet.

Genomförande:

1. Deltagarna klipper de åtta rutorna på bilden som de fått till lappar. Lägg först lapparna F, G och H åt sidan.
2. Varje deltagare ordnar lapparna A–E framför sig på bordet i ordningen **EDCBA**. Lapparna är alltså placerade så att det till vänster finns fler punkter och på lappen till höger endast en punkt.
3. Alla lapparna svängs så att den vita sidan syns (men de byter inte plats!).
4. Som uppvärmning börjar ni räkna talen från noll och uppåt. Handledaren skriver med tusch ett tal i enlighet med tiosystemet samt det binära talet som bildas av lapparna.

0 - Ni gör ingenting, för ni har bara tomma papperslappar framför er.
Det binära talet är 0 eller alla bitar som används 00000.

1 - vänd lappen längst till höger (A) så att punkten blir synlig.

En punkt är synlig. Det binära talet är 00001.

2 - lappen längst till höger räcker inte längre till! Vänd följande lapp mot vänster (B). Nu är tre punkter synliga, vänd alltså igen lappen A. Två punkter är synliga. Det binära talet är 00010.

3 - vänd lappen längst till höger igen så den är rätt vänd. Tre punkter är synliga. Det binära talet är 00011.

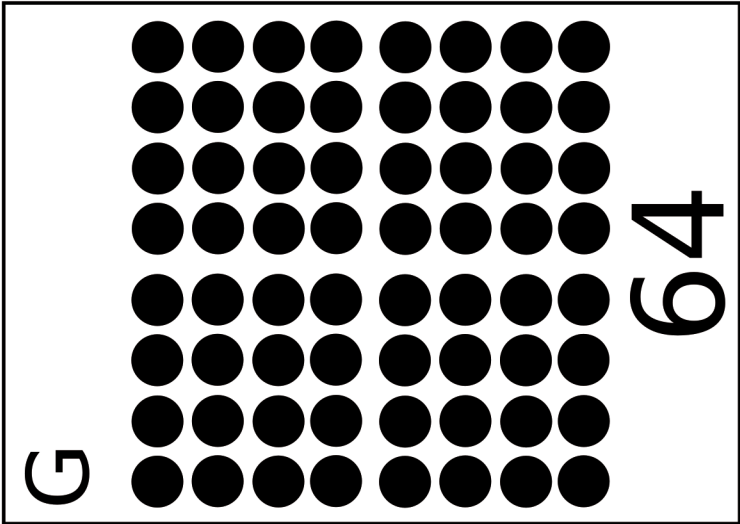
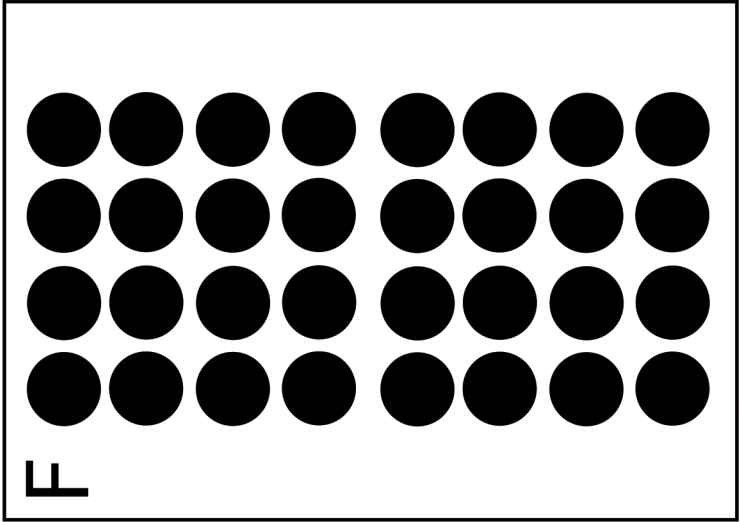
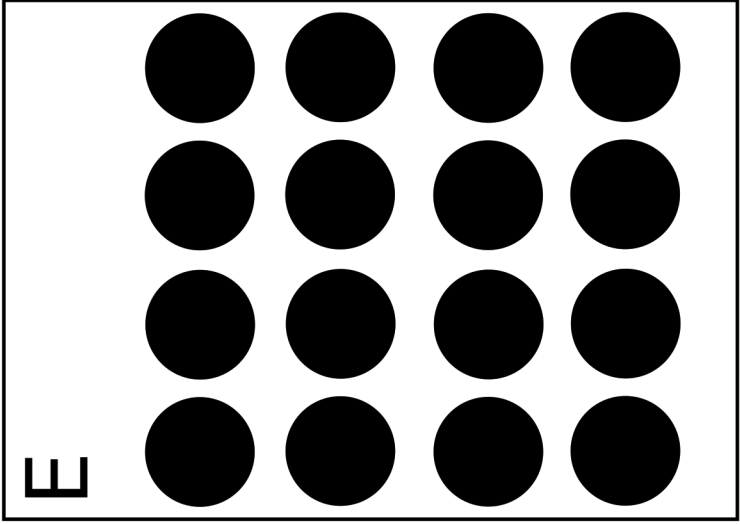
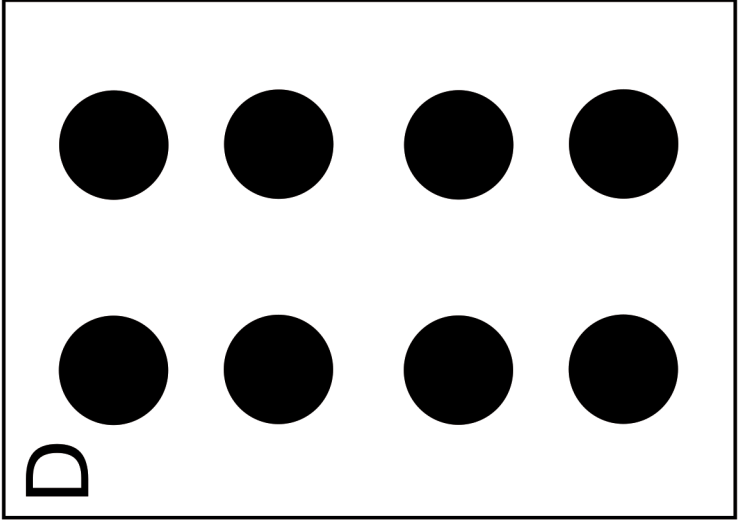
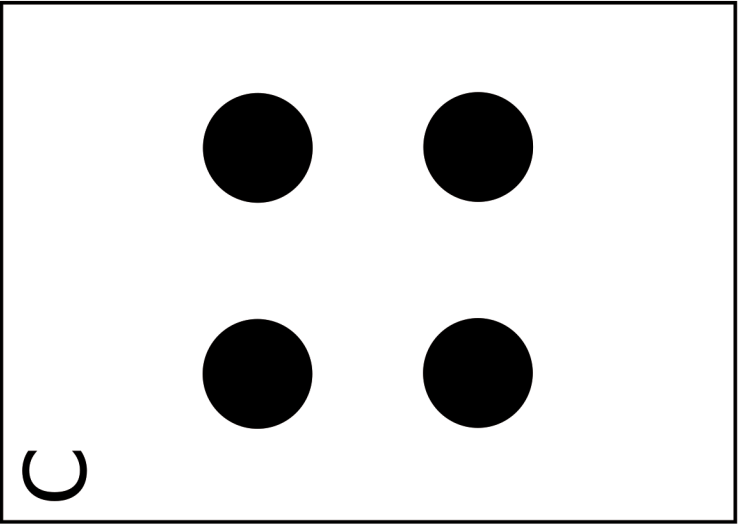
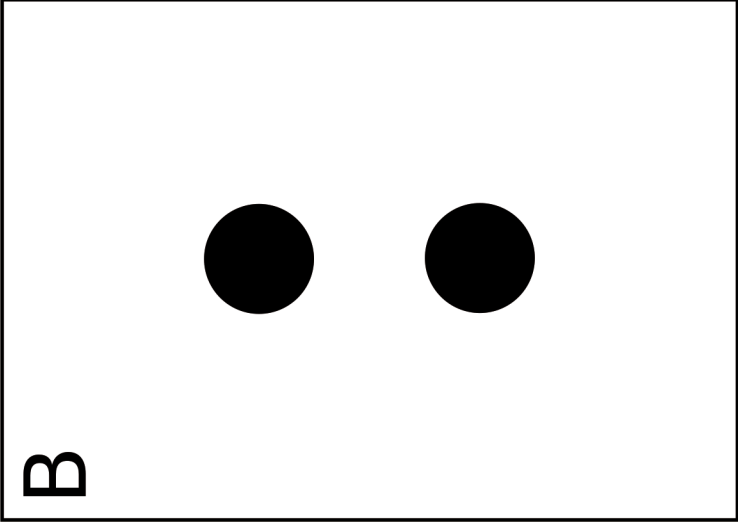
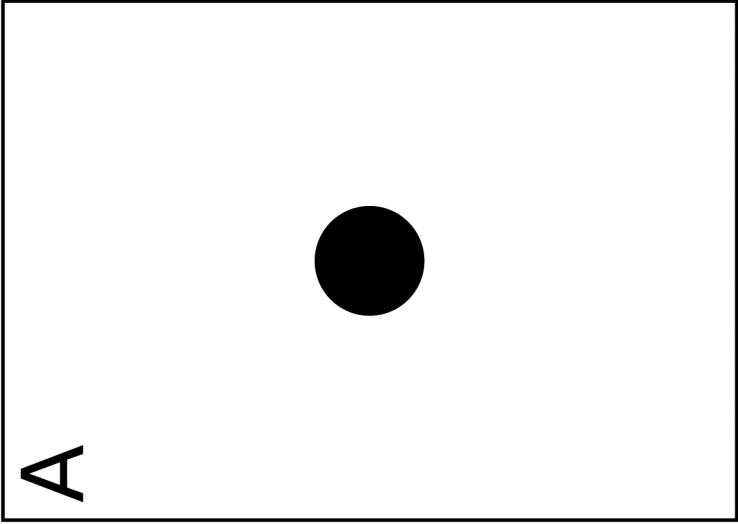
4 - ni måste igen ta en ny bit i användning! Vänd följande lapp mot vänster (C) upp. Nu är sju punkter synliga. Vänd alltså lapparna A och B upp och ner. Fyra punkter är synliga. Det binära talet är 00100.

Fortsätt så här tills deltagarna kan vända lapparna smidigt.

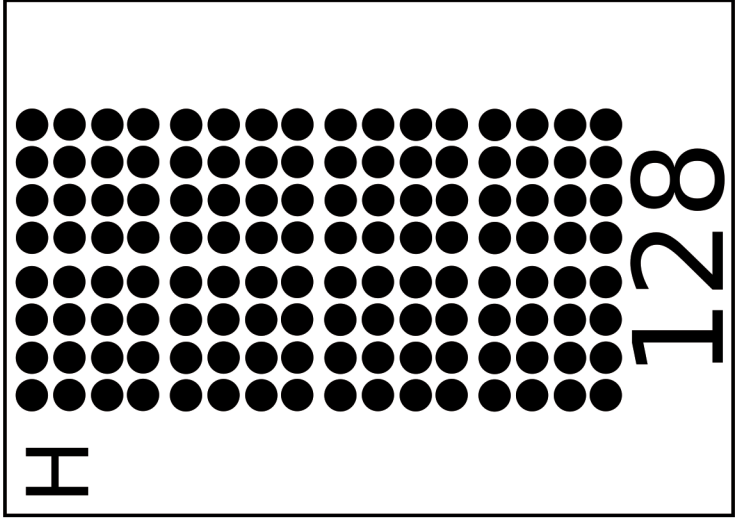
5. Fråga deltagarna vad de bekanta talen i tiosystemet är som binära tal. Till exempel 7, 11, 16 och till exempel deltagarens ålder.
6. Fråga deltagarna: hur skrivs talet 31 i det binära talsystemet? (Svar: 11111)
7. Fråga deltagarna: hur skrivs talet 32 i det binära talsystemet? (Svar: det behövs en ny bit! Sätt lappen F på vänstra sidan av de andra lapparna. På den finns 32 punkter, vänd alltså de övriga lapparna upp och ner. Det binära talet är alltså 100000.)
8. Om deltagarna orkar kan ni även ta lapparna G och H i användning, då lapparna är 8 till antalet, det vill säga det finns en byte utrymme. Byte är en grundenhet för datorer som anger till exempel storleken på hårddisken. En gigabyte är en miljard byte, det vill säga åtta miljarder bitar.

Nedan ännu en kontrolltabell över vissa tal i tiosystemet i binär form.

tiosystemet	binärt tal	tiosystemet	binärt tal	tiosystemet	binärt tal
1	00000001	11	00001011	30	00011110
2	00000010	12	00001100	40	00101000
3	00000011	13	00001101	50	00110010
4	00000100	14	00001110	60	00111100
5	00000101	15	00001111	70	01000110
6	00000110	16	00010000	80	01010000
7	00000111	17	00010001	90	01011010
8	00001000	18	00010010	100	01100100
9	00001001	19	00010011	128	10000000
10	00001010	20	00010100	255	11111111



64



128

Kluraklubbdiplom

Matematik & Programmering & Berättelser

har framgångsrikt avlagt utbildningens områden:

- ☐ Juveltjejer
- ☐ Havsäventyret
- ☐ Astronomerna
- ☐ Detektiverna
- ☐ Jordenruntresan





